JEDE AUSGABE MIT FIRMENREGISTER UND PRAKTIKUMSBÖRSE

04/2008 € 6,90

making games MAGAZIN

MAGAZIN FÜR SPIELE-ENTWICKLUNG UND BUSINESS-DEVELOPMENT

COMMUNITY MANAGEMENT

WARUM ES JEDEN BETRIFFT, WOMIT SIE EFFIZIENT STEUERN UND WIE SIE AUS LAUFKUNDEN TREUE MITGESTALTER MACHEN

DAS ENDE DER KONSOLEN

EINE PROVOKANTE ANALYSE DES DIRECTX-ERFINDERS

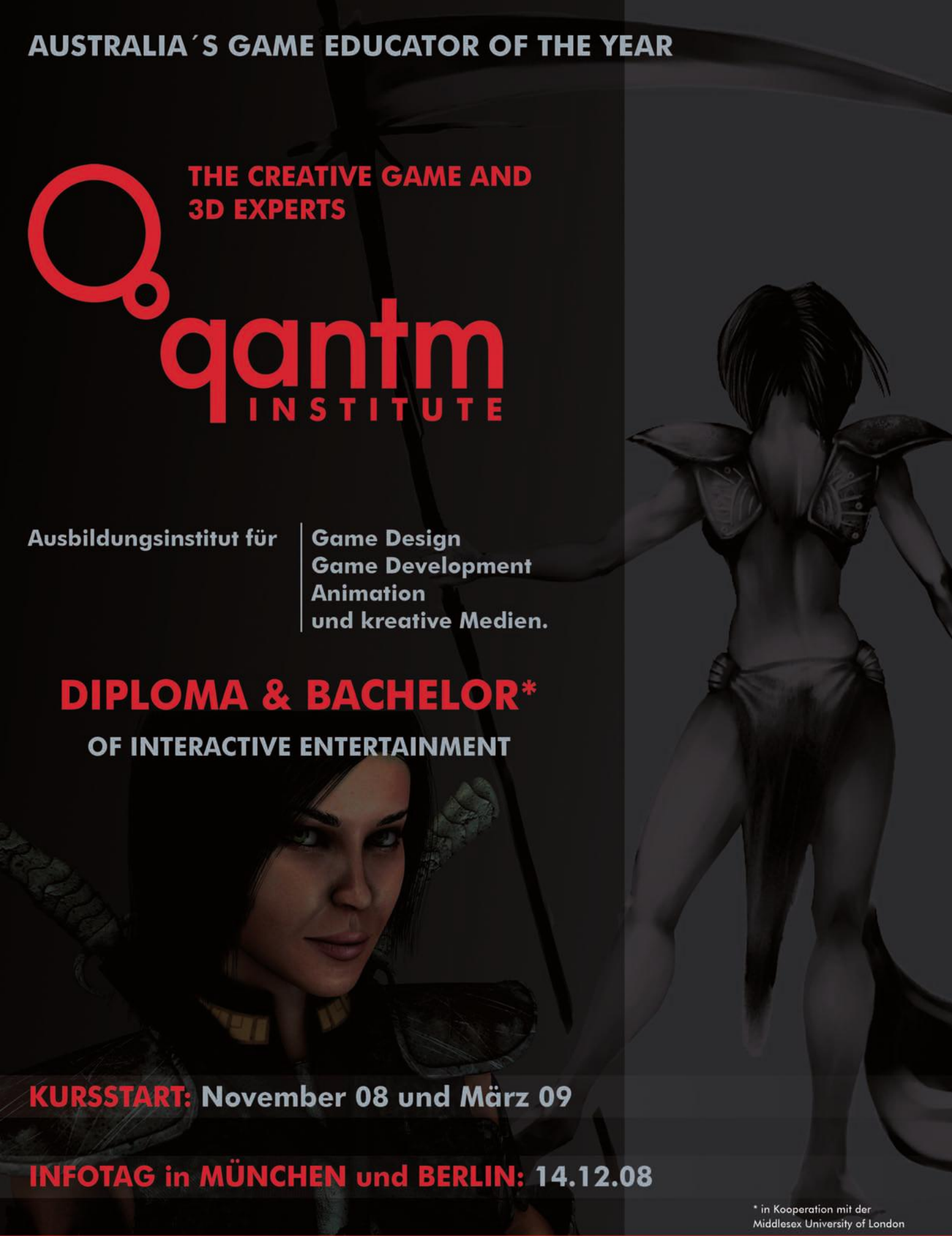
EXKLUSIV-INTERVIEW

IM GESPRÄCH MIT NINTENDO-LEGENDE SHIGERU MIYAMOTO

PROGRAMMIERUNG

WIE SIE IHR SPIEL PERFORMANT AUF DIRECT3D 10 PORTIEREN

4 197050 406909 04



AUSTRALIA´S GAME EDUCATOR OF THE YEAR

THE CREATIVE GAME AND
3D EXPERTS

qantm
INSTITUTE

Ausbildungsinstitut für

Game Design
Game Development
Animation
und kreative Medien.

DIPLOMA & BACHELOR*

OF INTERACTIVE ENTERTAINMENT

KURSSTART: November 08 und März 09

INFOTAG in MÜNCHEN und BERLIN: 14.12.08

* in Kooperation mit der
Middlesex University of London

www.qantm.de - www.qantm.tv

www.sae.edu

info@qantm.de

Berlin: +49 (0)30 49860054 - München:+49 (0)89 89068771

SAE INSTITUTE

Die Überlebensversicherung
Making Games Start

Heiko Klinge

ist Projektleiter vom Making Games Magazin.

So viel Selbstbewusstsein ist hoffentlich erlaubt: Wenn wir uns für ein Titelthema entscheiden, dann sind wir davon überzeigt, dass es nicht nur für einige Wenige, sondern für den Großteil der Branche relevant ist. Dennoch waren wir zugegebenermaßen recht ordentlich überrascht, als wir kurz vor Heftschluss sogar auf einem Nintendo DS verstärktes Community-Engagement entdeckten. So geschehen in »Sonic Chronicles: Die Dunkle Bruderschaft«, einem verhältnismäßig simplen Rollenspiel von Bioware. Das empfiehlt uns in seinen Hilfstexten eben nicht nur, dass wir grundsätzlich alle Dorfbewohner anquatschen sollen, sondern auch, dass wir für weitere Tipps und Tricks die Community-Seite zum Spiel besuchen sollen. Haben wir natürlich ausprobiert. Und tatsächlich: Dort bekommen wir in fünf Sprachen einen ausführlichen Strategie-Leitfaden, dürfen Fragen an die Entwickler stellen und können sogar einen Bonus-Gegenstand fürs Spiel freischalten.

Dass Bioware, seit jeher ein Vorreiter in Sachen Community Management, auch bei DS-Spielen so viel Aufwand in die Fan-Arbeit steckt, hat einen ganz simplen und fast schon enttäuschend nüchternen Grund: finanzielle Absicherung. In einem Gespräch auf der letzten GDC über die stetig wachsenden Entwicklungsbudgets verriet uns Firmenchef Ray Muzika, dass die Bioware-Community mittlerweile mehr als vier Millionen registrierte User zählt. Und mit einem Lächeln ergänzte er: »Wenn nur ein Zehntel davon Mass Effect kauft, sind wir profitabel. Kein allzu großes Risiko, oder?« In diesem Sinn: Viel Spaß mit unserer Titelstory!

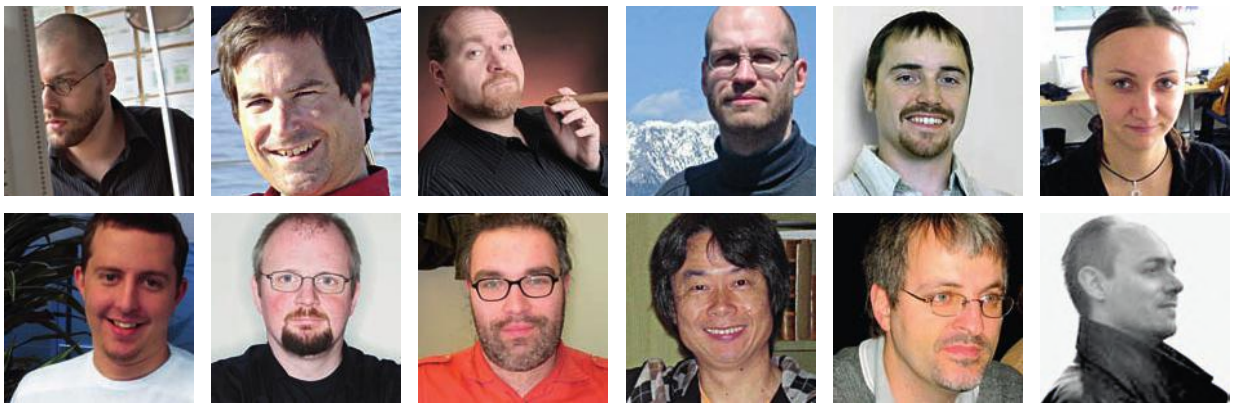
Achtung: Pathos!

Community Management betrifft natürlich nicht nur Spiele-Entwickler, sondern auch Heftmacher – und das ganz besonders beim Making Games Magazin. Denn es gibt wohl kaum eine andere Zeitschrift, die von ihren Lesern derart aktiv mitgestaltet wird – schließlich schreiben nicht wir die Artikel, sondern unsere »Community-Mitglieder«: Studenten und Manager, Berufsanfänger und Branchen-Veteranen, Grafiker und Programmierer. Ein paar davon sehen Sie wieder am unteren Seitenrand. Wir möchten uns an dieser Stelle ganz herzlich bei allen bedanken, die unser kleines Magazin im letzten Jahr mit Beiträgen, Feedback und Vorschlägen unterstützt haben. Und wir bedanken uns bei allen im Voraus, die uns ihre Meinung zu den Heften des Jahres 2008 an info@makinggames.de schicken: Welche Artikel haben Ihnen besonders gefallen? Was hätten wir uns vielleicht auch schenken können? Welche Themenschwerpunkte wünschen Sie sich für die kommenden Ausgaben? Wir hören auf unsere Community – versprochen!

Viel Spaß beim Lesen und
ein erfolgreiches Jahr 2009 wünscht

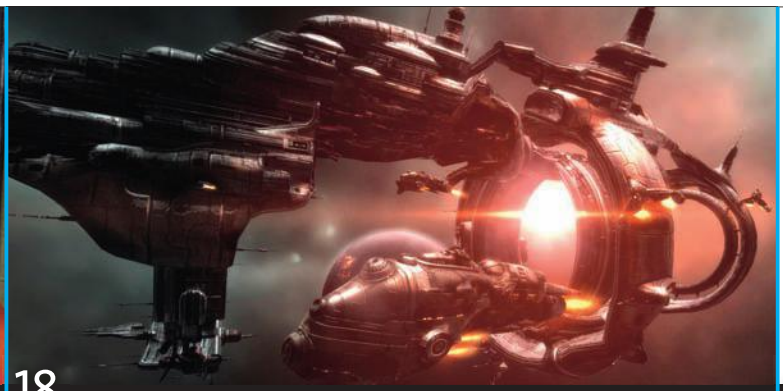
das Team von Making Games

»Community Management betrifft natürlich nicht nur Spiele-Entwickler, sondern auch Heftmacher – und das ganz besonders beim Making Games Magazin.«



04/2008
Inhalt

64 Im Gespräch mit Shigeru Miyamoto.



18 Der Lead Economist von CCP über die Spieler-Mitbestimmung in Eve Online.

46 Provokante Analyse: Warum

36 Programmierung: Warum Battleforge unter Direct3D 10 30 Prozent schneller läuft.

24 Wie Entwickler Crytek versucht, seine Fans langfristig zu binden.

Branche

06 **Branche / Köpfe**
Bemerkenswertes aus der Branche

07 **Eventkalender**
Wichtige Branchenveranstaltungen

Titelstory Community Management

8 **Best Practice** Community Management
von Nadja Bastawi und Thomas Lagemann

14 **Best Practice** In-Game-Support in MMOs
von Teut Weidemann

18 **Case Study** Eine symbiotische Beziehung
von Dr. Eyjo Gudmundsson

24 **Case Study** Die Crytek-Community

Rubriken

03 **Editorial**

80 **Ein Tag bei ...** EA Phenomic

82 **Vorschau**

82 **Impressum**

Grafik

26 **Workshop** Texturerstellung für ein 3D-Modell
von Chris Noeth

32 **Case Study**
2D-Animationen in A Whispered World
von Franziska Reinhard, Gunnar Bergmann
und Fabian Pahl

die Konsolen aussterben werden.

32 Case Study über die 2D-Animationen in A Whispered World.

08 Wichtige Regeln für Community Management.

80 Ein Tag bei ... EA Phenomic.

42 So funktioniert Designdokumentation.

68 Das Studentenprojekt Swordmaster.

Entwicklung

36 **Programmierung**
Portierung von Direct 3D 9 zu Direct 3D 10
von Ralf Kornmann

42 **Projektmanagement**
So funktioniert Designdokumentation
von Dirk Rieger

Business

46 **Essay** Das Ende der Konsolen
von Alex St. John

51 **Interview** Im Gespräch mit David Braben
von Heiko Klinge

54 **Studio Management**
Effiziente Projektstrukturen
von Erik Simon

58 **Kopierschutz** Hardware-Kopierschutz
von Steffen Kätsch

Campus

64 **Interview** Im Gespräch mit Shigeru Miyamoto
von Nino Kerl

68 **Studentenprojekt** Swordmaster
von Markus Ambrus und Rainer Lonau

72 **Ausbildung** Praktikumsbörse

Märkte

75 **Marktforschung**
GameStar- und GamePro-Leserdaten
von Heiko Klinge

76 **Firmenregister** Die Branche im Überblick

Die Nominierten

Bestes Deutsches Spiel 2008

Crytek
Crysis

Funatics
Die Siedler – Aufbruch der Kulturen

Radon Labs
Drakensang

Daedalic
Edna Bricht Aus

Animation Arts
Geheimakte 2: Puritas Cordis

Ascaron
Sacred 2: Fallen Angel

Bestes Deutsches Kinderspiel 2008

Brainmonster Studios
2weistein: Das Geheimnis des roten Drachen

WEB VV
Felix: Eine wundersame Reise durch die Zeit

Outline Development
Frag doch mal ... die Maus!

Morgen Studios
Playmobil Piraten: Volle Breitseite!

Independent Arts
Prinzessin Lillifee: Meine wunderbare Welt

Bester Soundtrack

Crytek
Crysis

Crytek
Cyrsis Warhead

Radon Labs
Drakensang

Animation Arts
Geheimakte 2: Puritas Cordis

Ascaron
Sacred 2: Fallen Angel

Beste Story & Spielewelt

Crytek
Crysis

Radon Labs
Drakensang

Daedalic Entertainment GmbH
Edna Bricht Aus

Animation Arts
Geheimakte 2: Puritas Cordis

Ascaron
Sacred 2: Fallen Angel

Bestes Deutsches Browserspiel

Chromatrix
Das Schwarze Auge: Verschwörung in Ferdok

Gameforge
Ikariam

Travian Games
Travianer

upjers
Wurzelimperium

Trollgames
Xhodon

Deutscher Entwicklerpreis: Die Nominierten

Sind Sie gerade unbeobachtet? Prima, dann klopfen Sie sich doch mal kurz selbst auf die Schulter. Der Deutsche Entwicklerpreis wird am 3. Dezember zwar schon zum fünften Mal verliehen, aber noch nie gab es so viele gute Gründe für unsere Branche, sich selbst zu feiern. Denn die hiesigen Entwickler haben dieses Jahr eben nicht nur qualitativ hochwertige Spiele produziert, sondern vor allem auch kommerziell erfolgreiche. So konnten alle sechs Nominierten für die Kategorie »Bestes Deutsches Spiel 2008« ausnahmslos die Top 10 der Charts entern, Drakensang und Sacred 2 schafften es sogar bis auf Platz 1. Ebenfalls erfreulich: Mit »Edna bricht aus« hat erstmals auch ein ehemaliges Independent-Projekt beste Chancen auf einen Preis. Als offizieller Medienpartner des Deutschen Entwicklerpreises drücken wir selbstverständlich allen Nominierten gleich fest die Daumen. Wir sehen uns am 3. Dezember in der Essener Lichtburg! Und wir lassen es krachen! **Heiko Klinge**

EIGA-Awards: Deutscher Nachwuchs vorn dabei

In diesem Heft wird von unseren Autoren regelmäßig die Ausbildungssituation der deutschen Spielebranche bemängelt – zu Recht. Dabei geht es jedoch in erster Linie um die Quantität. Qualitativ ist der deutsche Nachwuchs auch international ganz vorn mit dabei. Das bewies er eindrucksvoll bei den European Innovative Game Awards (EIGA), die am 6. November in Darmstadt von der Europäischen Kommission verliehen wurden, natürlich mit dem Making Games Magazin als Medienpartner. Gleich zwei Abschlussprojekte von Studenten der Berliner Games Academy schafften es auf die Nominiertenliste: Blake White und Somersault. Letzteres gewann sogar den zweiten Preis und musste sich nur dem niederländischen Serious Game Sharkworld geschlagen geben. Eine Riesenleistung angesichts von ebenso klangvoller wie kommerzieller Konkurrenz wie Heavenly Sword, Wii Fit und Crysis.

Gameforge: Stipendien für Spiele-Entwickler

Browsergames-Entwickler Gameforge hat eine umfangreiche Kooperation mit der SRH Hochschule Heidelberg abgeschlossen. Ab dem Sommersemester 2009 will das Karlsruher Unternehmen in drei Nachwuchstalente investieren und die kompletten Ausbildungskosten für ein Informatik-Studium mit dem Schwerpunkt Game-Entwicklung übernehmen – immerhin 520 Euro im Monat, drei Jahre lang. Dementsprechend absolvieren die Stipendiaten ihr Praxis-Trimester natürlich auch direkt bei Gameforge. Außerdem finanziert Gameforge einen neuen Game-Entwickler-Medienraum an der SRH Hochschule, der mit 30 PCs ausgestattet wird. Interessierte können sich ab sofort direkt bei Studiengangsleiter Professor Tobias Breiner für die Stipendien bewerben: tobias.breiner@fh-heidelberg.de

Vier Fragen an ...

Klaas Kersting

ist Gründer und Vorstand der Gameforge AG.

Making Games Was muss ein Bewerber mitbringen, um Chancen auf ein Stipendium zu haben?

Klaas Kersting Der ideale Bewerber für ein Stipendium bei Gameforge bringt ein hohes Maß an Kreativität und natürlich große Begeisterung für Spiele mit. Wirklich gute Spiele programmieren sich nicht von selbst. Wir brauchen kreative Köpfe, die uns mit ihren Fähigkeiten dabei helfen, unsere Games fit für den Weltmarkt zu machen. Ist der Bewerber darüber hinaus flexibel, belastbar und offen für Neues, dann passt er perfekt zu uns.

Making Games Mit welchen Bedingungen ist das Stipendium verbunden?

Klaas Kersting Alle Stipendiaten haben selbstverständlich die Möglichkeit, nach dem Studium bei uns weiter zu arbeiten. Auf eine Verpflichtung bestehen wir aber nicht. Aus unserer bisherigen Erfahrung wissen wir jedoch, dass unsere Praktikanten die Vorzüge von Gameforge als Arbeitgeber erkennen und deshalb ohnehin weiter bei uns tätig sein möchten.

Making Games Warum haben Sie sich für eine Kooperation mit einer privaten Hochschule entschieden statt für eine Zusammenarbeit mit einer öffentlichen Universität?

Klaas Kersting Ganz entscheidend für die Kooperation mit der SRH ist natürlich der auf Game-Entwicklung spezialisierte Studiengang. Dort sitzen genau die Leute, die wir auch in Zukunft für unser Unternehmen benötigen. Zudem wird die Umsetzung unserer Ziele dank der flexibleren Strukturen einer privaten Hochschule besser gelingen als mit einem großen Universitätsapparat im Hintergrund.

Making Games Ihre Prognose: Wie wird sich die Ausbildungssituation bei Spieleentwicklern in den nächsten Jahren verändern?

Klaas Kersting Der Bereich Spieleentwicklung ist Zukunftsmarkt und Jobmotor zugleich. Es werden immer mehr Spezialisten gefragt sein. Ich bin sicher, die Hochschulen und Universitäten werden diesen Trend bald erkennen und dementsprechend ihre Lehrpläne anpassen. Gameforge sucht aber schon jetzt jede Menge gute Leute - und deshalb investieren wir auch schon jetzt.

Eventkalender

NOVEMBER 2008

Browsergames Forum

Ort: Frankfurt, Japan Tower
Termin: 22. bis 23. November 2008
Thema: Fachkonferenz
Kosten: freier Eintritt für Browsergames-Entwickler, sonst 200 Euro
Homepage: www.bgf2008.de

IGDA Chapter München

Ort: München, Augsburger Str. 8
Termin: 26. November
Thema: Regio-Summit, Branchen-Treff
Kosten: Eintritt frei
Homepage: www.igda.org/munich

DEZEMBER 2008

Gamecity Treff

Ort: Hamburg, 13. Stock
Termin: 2. Dezember 2008
Thema: Regio-Summit, Branchen-Treff
Kosten: Eintritt frei
Homepage: www.gamecity-hamburg.de

Deutscher Entwicklerpreis

Ort: Essen, Lichtburg
Termin: 3. Dezember 2008
Thema: Preisverleihungs-Gala
Kosten: 36 bis 84 Euro
Homepage: www.deutscher-entwicklerpreis.de

JANUAR 2009

Spieleentwicklertreffen NRW

Ort: Düsseldorf, Mediadesign Hochschule
Termin: 22. Januar
Thema: Vorträge und Networking
Kosten: Eintritt frei
Homepage: www.spieleentwicklertreff-nrw.de

FEBRUAR 2009

Mobile World Congress

Ort: Barcelona, Fira Montjuïc
Termin: 16. bis 22. Februar 2008
Thema: weltweit größter Mobile-Fachkongress
Kosten: 599 bis 3.799 Euro
Homepage: www.mobileworldcongress.com

MÄRZ 2009

Game Developers Conference

Ort: San Francisco, Moscone Center
Termin: 23. bis 27. März
Thema: weltweit größte Entwicklerkonferenz
Kosten: 175 bis 2.200 Dollar
Homepage: www.gdconf.com

Veranstalten Sie mit!

Sie organisieren eine Podiumsdiskussion oder planen eine Party und möchten, dass die Branche davon erfährt? Dann schreiben Sie uns an **info@makinggames.de, Stichwort »Eventkalender«** und wir veröffentlichen Ihre Veranstaltung in unserer nächsten Ausgabe.

Immer ehrlich zu den Fans
Community-Management

Thomas Lagemann und Nadja Bastawi vom Two Pi Team definieren, was CM ist, wie man CM betreiben sollte, wo die gefährlichsten Fehlerquellen liegen und welche Anforderungen CM erfüllen muss.

Nadja Bastawi (Dipl. Oec)

ist Inhaberin und Geschäftsführerin von Two Pi Team.

Thomas Lagemann

ist genau das Gleiche wie Nadja.

Two Pi Team wurde im Jahr 2005 gegründet und hat sich auf die besonderen Dienste rund um Computerspiele spezialisiert. Das Team bietet professionelle Forenmoderation und -betreuung sowie Community Management und weitere Leistungen in diesem Bereich. Zu den bekanntesten Kunden gehören Codemasters, Funcom und dtp.

Das Community Management sollte als zentrale **Schnittstelle** verstanden werden.

Wir beginnen bei den Basics – was ist Community Management überhaupt? Der Begriff »Community« kommt aus dem Englischen und bezeichnet grob eine Gemeinschaft, die mindestens ein gemeinsames Interesse teilt und eine eigene Identität aufbaut. Ein »Manager« ist in der allgemeinen Wirtschaft eine Führungskraft. Im Sport oder im künstlerischen Bereich ist der Manager ein Agent, der seinem Mandanten insbesondere die organisatorischen und geschäftlichen Belange abnimmt. Also:

Ist der Community Manager Führungskraft?
Er sollte es sein – innerhalb der Unternehmensstruktur. Aber auch für die Community?
Ist der Community Manager ein Agent?
Das sollte er sein – für die Community. Wie sich das mit seiner Aufgabe innerhalb des Unternehmens vereinbaren lässt, bleibt zu klären.

Community-Arbeit für Age of Conan auf einer **Messe**.

Eine Community besteht weder aus Untergebenen noch aus Zahlen oder materiellen Dingen. Es sind Menschen, Konsumenten, Endabnehmer des Produktes, die vielleicht in Zahlen ausgedrückt werden können – allerdings niemals fix, sondern stets variabel und immer mit mindestens einer Unbekannten. Einen Community Manager als reinen Vorgesetzten der Community zu betrachten und ihn so agieren zu lassen, wäre ein grobes Missverständnis in der Auslegung des Begriffes. Er muss eine Respektsperson sein, von der sich die Community etwas sagen lässt.

Wie sieht es aus mit dem Begriff der Community? Wir bewegen uns hier im speziellen Be-

reich der Konsumenten von PC-, Video- und Konsolenspielen. Ergibt also eine Spezialisierung des Begriffes, zum Beispiel in Richtung »Online Community«, einen Sinn? Auf den ersten Blick ja, denn viele Bereiche der Community-Arbeit finden über das Medium statt, dem sich die meisten der Spieler verbunden fühlen: dem Internet. Dennoch wäre eine solche Spezialisierung des Begriffes nicht unbedingt korrekt, da man sich zu sehr auf ein einziges Medium versteifen würde. Im Internet sind die Community-Mitglieder sichtbar, die sich aktiv beteiligen. Wo sind die unsichtbaren Mitglieder, wie erreicht man sie?

Viele der unsichtbaren Mitglieder sind tatsächlich auch im Internet zu finden, allerdings halten sie sich passiv im Hintergrund und beteiligen sich nicht an Diskussionen und Aktionen. Falsch wäre es daher, anzunehmen, dass es sie nicht gibt. Es gibt sie, und ihre Zahl ist groß, sehr groß, gemessen an den aktiven, sichtbaren Mitgliedern. Einige der unsichtbaren Mitglieder sind nur für die offiziellen Plattformen unsichtbar und vermeintlich inaktiv, jedoch sehr präsent auf ihren eigenen, privaten Plattformen, wie zum Beispiel Gildenseiten, Social Networks und Fanseiten, die aufgrund von Spielerinitiativen ins Leben gerufen werden. Weitere ziehen ausschließlich Medien vor, die stofflicher sind als das Internet – zum Beispiel die klassischen Printmedien oder Veranstaltungen wie Messen, Roadshows, Conventions und so weiter.

Das Community Management sollte alle diese Medien und Treffpunkte der Community innerhalb und außerhalb des Internets pflegen. Das gilt auch und gerade für spätere Phasen im Lebenszyklus eines Spiels.

Interaktion zwischen Marketing und Community Management

In den vorangegangenen Absätzen tauchen Wörter wie »Medien« und »Lebenszyklus« auf. Begriffe, die klassischerweise mit den Aufgaben des Marketing verknüpft sind, denn das Marketing ist für die Vermarktung eines Produktes zuständig. Wozu wird dann überhaupt ein Community Management gebraucht? Ist nicht das Marketing dafür verantwortlich, dass das Produkt läuft? Natürlich. Aber in der heutigen Zeit, insbesondere in modernen, innovativen Marktsegmenten wie dem der PC-, Video- und Konsolenspiele reicht eine einfache Vermarktung nicht mehr aus, um Kunden zu locken und vor allem, um sie zu binden – sei es an das Produkt oder gar an das entwickelnde Unternehmen.

Die Kunden von heute wollen Kommunikation. Eine Abteilung entwickelt die Vermarktungsstrategien, die andere kommuniziert sie. Beide Abteilungen haben viele Berührungs- und dadurch auch Brennpunkte. Niemand ist näher an den Endkonsumenten als das Community Management. Die Community Manager sind es, die die Stimmungen der Community auffangen, die ihre Belange, Wünsche und Kritik kennen und die wissen, wie die Community auf bestimmte Dinge reagiert.

Community-Pflege auf einer Messe: Die **Zielgruppe** wartet auf frische Informationen.

Keine noch so gute Marketingkampagne erreicht ihr Ziel, wenn sie zum falschen Zeitpunkt oder an die falschen Personen adressiert wird. Die Devise zwischen den Abteilungen sollte sein: Reden Sie miteinander. Insbesondere das Marketing sollte sich ins Bewusstsein rufen, wie wichtig das Community Management mit seiner Nähe zur Zielgruppe ist. Fragen Sie, wann ein geeigneter Zeitpunkt für eine Kampagne ist. Fragen Sie, was die Community wirklich will.
Eine im Vorfeld erhobene statistische Erhebung zu der potenziellen Zielgruppe eines noch zu realisierenden Spiels muss nicht mit der Zielgruppe übereinstimmen, die Ihnen Ihr Community Manager nennt. Sie haben Zahlen. Er kennt das Spielerverhalten und die Bedürfnisse der Spieler. Und ringen Sie sich zumindest ab und zu dazu durch, dass das Community Management die Ankündigungen, Anzeigen und sonstigen Kampagnen-Texte selbst formulieren darf. Texte, die viele Superlative enthalten, mögen probate Marketing-Mittel sein – der Community Manager weiß, ob die Community das überhaupt hören will, ob sie nicht von solchen Ankündigungen übersättigt ist oder ob ein solcher Text, zu einem ungünstigen Zeitpunkt veröffentlicht, nicht gar einen Effekt hat, der sich komplett konträr zu seiner ursprünglichen Intention auswirkt.

Das Wichtigste an der Interaktion ist jedoch, dass sie überhaupt stattfindet. Informieren Sie als Marketing Manager Ihren Community Manager rechtzeitig über die bevorstehenden oder angedachten Kampagnen. Und als Community Manager sollten Sie die Marketing-Abteilung rechtzeitig und aussagekräftig über Stimmungen in der Community informieren!

Internationale Communitys

Noch schwieriger wird gerade diese Interaktion bei einer Community, die international ausgelegt ist. Jede Kultur hat unterschiedliche Bedürfnisse; nicht nur in Bezug auf ein Produkt, sondern auch in Bezug auf die Kommunikation. Das oben genannte Beispiel zu Texten, die viele Superlative enthalten, trifft eher auf die deut-

Die Community **organisiert** sich im Spiel selber.

sche Community zu. In England und den USA hingegen verfehlt dieses Stilmittel seine Wirkung seltener. Es kommt auf die Dosierung an, denn allzu schnell fühlt sich eine Sprachgemeinschaft einer anderen nachgeordnet.

Das Community Management sollte in der Lage sein, zu erkennen, wo das richtige Maß liegt. Zusätzlich obliegt es ihm, die Ankündigungen mit einer zusätzlichen Kommunikation zu begleiten, genau aus dem Grund, Missverständnisse in Bezug auf die Gewichtung der unterschiedlichen Sprachgemeinschaften auszuräumen. Dies gilt im Besonderen für die Kommunikation von Informationen.

Ein regelrechter Informationshunger ist allen Sprachgemeinschaften gemein. Sie alle wollen, dass man mit ihnen redet, dass sie auf dem Laufenden gehalten werden und dass sie Rückmeldungen auf ihre Fragen, Vorschläge, Kritik und Anmerkungen bekommen. Ein Community Manager sollte dafür sorgen, dass eine wichtige Information simultan oder zumindest sehr zeitnah in allen Sprachbereichen erhältlich ist. Dies ist eine knifflige Aufgabe, und sie wird umso schwerer, wenn das Unternehmen selbst international ausgerichtet ist. So kann zum Beispiel ein enthusiastischer, englischsprachiger Mitarbeiter eines Unternehmens die deutschsprachige Community extrem verstimmen, wenn er immer mal wieder in dem englischsprachigen Teil des zum Spiel gehörigen offiziellen Forums einige Statements abgibt, ohne dies dem Community Manager mitzuteilen.

Es ist dem Community Manager nahezu unmöglich, diese Aussagen immer zeitnah zu bemerken und in die anderen Sprachbereiche zu tragen. Von kontinentalen Zeitverschiebungen reden wir hier besser erst gar nicht. Die Folge ist, dass sich die anderen Sprachbereiche als »Fans zweiter Klasse« behandelt sehen – ein Eindruck, der oft gerade dann verstärkt wird, wenn es sich um ein Unternehmen handelt, das zum Beispiel die Realisierung von Aktionen rund um das Spiel nicht simultan stattfinden lässt.

Informationspolitik

Die Beschaffung von Informationen für die Community ist somit eine der wichtigsten Aufgaben des Community Managements. Voraussetzung dafür ist, dass das Community Management diese Informationen erhält. Zu oft sieht die Praxis jedoch leider so aus, dass der Community Manager wichtige Informationen selbst erst aus der Presse erfährt, zum Beispiel aus Exklusiv-Interviews, die ein anderer Mitarbeiter des Unternehmens gegeben hat.

Eine Community besteht aus vielen Individuen. Zusammen sind sie unschlagbar, was die Beschaffung von Informationen betrifft. Weiß der Community Manager nichts von einer Information, so wird er oft erst durch die Community damit konfrontiert, oftmals in Form von aus dem Zusammenhang gerissenen Zitaten ohne Quellenangaben. Er muss nun einen Teil seiner wertvollen Zeit damit verbringen, herauszufinden, ob es sich um eine wirklich offizielle Information handelt und von wem sie in welchem Zusammenhang stammt. Erst dann kann er der Community gegenüber die Information verifizieren und weiterführende Fragen beantworten. Handelt es sich um eine heikle Information (etwa: die Verschiebung des Spiels), so muss er ansonsten ohne Hintergrundwissen dem Unmut der Community entgegenwirken.

Untrennbar damit verbunden ist die Art und Weise, wie mit Informationen umgegangen wird. Ein Unternehmen muss dazu unbedingt eine einheitliche Philosophie entwickeln. Es ist schlimm, wenn eine Information auf unterschiedliche Weise dargestellt wird. Noch schlimmer ist es, wenn die Community den Eindruck erhält – und den erhält sie schnell – dass etwas unehrlich kommuniziert wird. Ehrlichkeit, beziehungsweise Offenheit, sollte in jeder Überlegung zur Findung einer Informationsphilosophie enthalten sein. Der schlimmste Fall jedoch ist, gar keine Informationen herauszugeben.

Vertrauen Sie Ihrem Community Management also Informationen an, auch wenn diese der Community noch nicht mitgeteilt werden dürfen. Vertrauen Sie dem Community Management, dass es die richtigen Worte findet, die Community wissen zu lassen, dass es Informationen wenigstens gibt, sie aber noch nicht für deren Ohren bestimmt sind.

Aufbau des CM

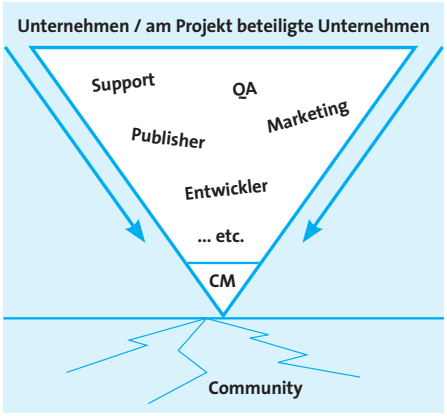
Wir verwenden bewusst keine einheitliche Terminologie. Wir reden von dem Community Manager, dem Community Management und einfach einer »Abteilung«, ungeachtet, ob es sich um interne oder externe Personen handelt. Alle diese Strukturen sind möglich, und alle haben ihre Vorzüge. Zu empfehlen ist dabei stets die Bildung eines Teams, wie aus den weiterführenden Erläuterungen noch deutlich wird. Mindestens eine Person sollte jedoch immer vor Ort im Haus des Unternehmens sitzen und von dort aus arbeiten. Alle Weiteren können außerhalb des Unternehmens angesiedelt sein. Zudem sollte man bedenken, dass die Arbeitszeiten eines Community Managers nicht unbedingt nach dem herkömmlichen Arbeitstag gestaffelt werden können. Eine Community ist keine Stechuhr, die ihre Fragen nur zu bestimmten Zeiten stellt und nur zu vordefinierten Zeiten Aufmerksamkeit benötigt.

CM und Unternehmensstruktur

Wie kann das Community Management also in das Unternehmen eingegliedert werden? Das Community Management sollte, wenn man die Struktur des Unternehmens grafisch darstellt, in der Schnittmenge sitzen, mit Kontaktmöglichkeiten zu nahezu allen anderen Bereichen, ohne sich hier auf eine zu hierarchische, sprich nachgelagerte, Struktur zu versteifen. Auch hier sind Interaktion und Informationsfluss der Schlüssel. Wir geben zu, dass diese Eingliederung in eine Unternehmensstruktur im Mo-

ment eher selten anzutreffen ist und eine Idealvorstellung darstellt. In der Praxis sieht es häufig so aus, dass es sich bei der Struktur um eine auf dem Kopf stehende Pyramide handelt, wobei das Community Management die unten liegende Spitze der Pyramide darstellt, auf die von vielen, vermeintlich darüber liegenden Abteilungen, Druck ausgeübt wird. Das liegt nicht zuletzt an den Missverständnissen, die in Bezug auf die Arbeit des Community Managements noch vielerorts herrschen. Wie können diese Missverständnisse entstehen?

Der Begriff des Community Managers ist zumindest hierzulande noch sehr neu und daher nicht ausreichend definiert. So wird er mit vielen Aufgaben in Verbindung gebracht, die sehr selektiv im Auge des Betrachters liegen.



So nicht: Auf das CM wird von »oben« **Druck** ausgeübt.

Marketing und Community Management:
Das Marketing sieht das Community Management oft als reine »ausführende Gewalt« an. Informationen erhält das Community Management erst, wenn es fertige Texte bekommt. Die wertvollen Ressourcen des Community Managements z.B. als Informationslieferant aus der Spielerbasis werden hingegen oftmals verkannt.

Support und Community Management:
Diejenigen, die mit dem Support der Spieler beauftragt sind, sehen das Community Management als willkommene Unterstützung, weil es oft in Foren auf erste Fragen antwortet und somit das Aufkommen von Supportanfragen vermindert. Leider wird nicht wahrgenommen, um wie viel die Anfragen vermindert werden könnten, und es gibt auch nur selten im Umkehrschluss die Möglichkeit, das Community Management an den Erkenntnissen der Supportabteilung zu beteiligen. Es kommt nicht selten vor, dass einem Spieler geholfen wird, es aber für den Community Manger keine Möglichkeit gibt, auf die Problemlösung zuzugreifen, um weitere Fragen zu beantworten. Der Zugang zu umfangreichen Wissensdatenbanken fehlt oft.

Qualitätssicherung und Community Management:
Die Qualitätssicherung kennt den Wert des Community Managements als wichtigen Informationsbeschaffer. Das Community Management bekommt, neben technischen Meldesystemen von Fehlern, über die direkte Kommunikation mit der Community als Erstes mit, wenn etwas nicht funktioniert oder die Community mit einer Funktion des Spiels unzufrieden ist. Umgekehrt erfährt das Community Management oft erst dann von Verbesserungen oder ausgebesserten Fehlfunktionen, wenn es zum Beispiel die sogenannten Patchnotes zur Übersetzung oder zur Reinschrift bekommt.

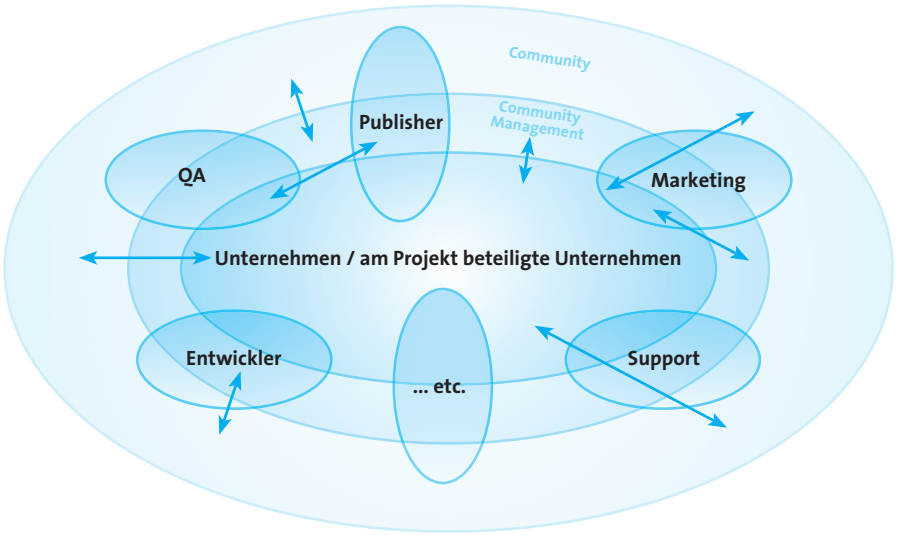
Entwicklerstudio / Publisher und Community Management:
Diese, von der Community als »die Offiziellen« bezeichneten, Personengruppen sehen in dem Community Management ebenfalls einen Informationslieferanten und betrachten es zudem als Kanal für die Umsetzung der Unternehmensinteressen. Von dem Community Management gesammeltes Feedback aus der Community wird gern gesehen. Am liebsten in Form von Diagrammen und Zahlen, denn die Zeit, lange Texte zu lesen, hat niemand. Entsprechende vom Community Management hinzugefügte Empfehlungen und manchmal auch Warnungen werden jedoch zu oft übergangen und erst dann bemerkt, wenn es zu spät ist.

Ein Beispieldialog:
Publisher: »Wir werden für das Produkt Preis AB festsetzen.«
CM: »Erstes Feedback aus der Community: Es scheint ein wenig hoch; ist die versprochene Sonderleistung schon enthalten?«
Publisher: »Nein.«
CM: »Das sollten wir ihnen rechtzeitig sagen.«
Publisher: »Ach was, wird schon gut gehen.«

Die Folge: Krieg im Forum. Und der Community Manager, der gewarnt hat, würde am liebsten auf eine einsame Insel flüchten. Nahezu jeder in einem Unternehmen hat also eine gewisse Erwartungshaltung gegenüber dem Community Management, und jeder setzt diese Abteilung gerne für seine Belange ein. Das ist auch durchaus legitim – wenn man dabei nicht vergisst, dass ein Community Manager auch nur einen Kopf und zwei Hände hat, um all diese Anforderungen in die Tat umzusetzen.

Man sollte ihn auch nicht als das Ende der eigenen Kette ansehen, sondern als Bindeglied zwischen der Abteilung und der Community. Und ein Bindeglied funktioniert am besten, wenn es in beide Richtungen weisen kann. Der Community Manager sollte nicht nur reiner Befehlsempfänger sein. Auch ihm steht ein gewisses Maß an Weisungsbefugnis durchaus gut zu Gesicht.

Zudem muss darauf geachtet werden, dass der Community Manager nicht in einen Interessenkonflikt gerät. Welchen Wunsch erfüllt er nun zuerst? Wo liegen die Prioritäten? Darf er das selbst entscheiden? Und bei alledem darf er sein Hauptziel nicht aus den Augen verlieren: für die Community da zu sein.



Der Idealfall: Alle Unternehmensbereiche **kommunizieren** mit der Community und untereinander.

Tägliche Gratwanderungen

Der letzte Absatz sollte verdeutlicht haben, dass der Community Manager ständig vor Entscheidungen steht. Sein gesamtes Tagwerk besteht aus Gratwanderungen, die es zu meistern gilt. Viele Community Manager kommen direkt aus der Community. Viele Mitarbeiter aus den unterschiedlichsten Bereichen eines Unternehmens in der Computerspiele-Branche sollten sich nicht mehr nur der Frage stellen: »Haben Sie jemals ein Spiel gespielt?«, sondern auch eine Antwort haben auf: »Waren Sie je Mitglied einer Community?« Man sollte wissen, wie eine Community denkt, agiert und funktioniert, denn auch, wenn sie aus Individuen besteht, ist sie auch eine Einheit, die sich durch besondere Verhaltensmuster und Denkweisen auszeichnet. Der Community Manager muss sich also täglich fragen, ob er einer der ihren oder einer von denen ist. Im Zweifel ist er für die jeweilige Partei sowieso immer auf der falschen Seite. Er darf die Belange der Community nicht aus den Augen lassen, denn sonst wird er der Community gegenüber unglaubwürdig. Auf der anderen Seite muss er unternehmerisch denken können und auch innerlich fest hinter gewissen Regeln und Fakten stehen. Der Community Manager muss also dem Unternehmen die Wünsche der Community nahe bringen und der Community die Notwendigkeit von wirtschaftlichen Entscheidungen. Und dabei die Balance schaffen, von beiden »Parteien« ernst genommen und respektiert zu werden. Ein Community Manager, der mit dem Erlangen seiner Position vergisst, dass er einst aus einer Community kam und sich nur noch als Manager im Sinne von Führungsperson sieht, kann das Vertrauen der Community verlieren. Ein zu kumpelhaft agierender Community Manager kann hingegen den Respekt der Community verlieren.

Das Zauberwort heißt »Grenzen ziehen«. Ein schwieriges Unterfangen, denn geht man einen Schritt weiter, so erkennt man, dass diese Grenzen niemals starr sein dürfen, sondern für viele Bereiche flexibel, dehnbar sein sollten. Hierzu bedarf es eines Beispiels:

In einem Forum gelten Regeln, denn ohne sie würde Chaos herrschen, und viele Spieler würden abgeschreckt: Die Community stagniert oder entwickelt sich größenmäßig zurück, denn gerade die unsichtbaren Mitglieder, die sich nicht aktiv beteiligen, werden abgeschreckt. Das Community Management sollte mit darauf achten, dass die Regeln eingehalten werden. Es stehen gewisse Werkzeuge zur Verfügung, zum Beispiel das Schließen von Themen. Nehmen wir an, das Schließen von bestimmten Themen wird rigoros durchgesetzt, um die Regeln einzuhalten. Nun ergibt sich aber etwas, das für starken Unmut innerhalb der Community sorgt. Die Community macht sich in Themen, die normalerweise geschlossen werden, Luft und versucht, sich Gehör zu verschaffen.

Der Community Manager kann nun hingehen und die Themen wie gewohnt schließen. Er kann sich aber auch bedingt auf die Seite der Community stellen und an diesem Tag diese Themen offen halten und nur darauf achten, dass es einigermaßen gesittet zugeht. Die Community wird es ihm danken. Der Community Manager muss nun aber damit rechnen, sein Verhalten dem Unternehmen gegenüber zu rechtfertigen, und er muss der Community zu verstehen geben, warum er am nächsten Tag diese Themen wieder rigoros schließt.

Spätestens hier muss er genau auf seine Aktionen und seine Wortwahl achten, denn ansonsten sieht er sich in einem Teufelskreis aus Diskussion und Rechtfertigung. Dumm sterben lassen darf man seine Community aber auch nicht. Also: Immer, wirklich immer, auf jedes Wort achten, sich überlegen, wie es ankommt und mögliche Konsequenzen durchspielen. Und durchaus mal einen Fauxpas zugeben.
Aber darf er das überhaupt? Gilt eine Entschuldigung als Schuldeingeständnis, das viel weitere Kreise zieht, als man bedacht hat? Fällt es gar auf das Unternehmen zurück?

Ähnliches ist dann der Fall, wenn der Community Manager sich mit Entscheidungen anderer Abteilungen konfrontiert sieht, bei denen es sich in den Augen der Community um Fehlentscheidungen handelt. Extrem wird es, wenn der Community Manager die Ansichten der Community sehr gut nachvollziehen kann. Dafür ist es gut, wenn der Community Manager weiß, wie es zu der Entscheidung kam. Das hilft ihm eventuell, diese zu verstehen, auch wenn er sie vielleicht immer noch nicht gut heißt. Allerdings wird er ganz andere Worte finden, um mit der Community zu kommunizieren.

Die nächste Gratwanderung ist die Gefahr des Machtmissbrauchs. Ein Community Manager hat eine gewisse Form der Macht über eine Community. Er kann sie lenken und leiten und sollte dies auch tun, jedoch stets neutral und im Sinne aller, also im Idealfall im Sinne des Pro-

duktes UND der Community, jedoch nicht in seinem persönlichen Sinne. Ein Community Manager sollte also neutrale Argumente für sein Handeln in der Community darlegen können.

Eine Community kann durchaus nervenaufreibend sein. Hier ist es von großem Vorteil, wenn der Community Manager kein Einzelkämpfer ist, sondern ein Team hinter oder um sich hat. Er kann dann mit anderen Team-Mitgliedern reden, um seine Neutralität zurückzugewinnen oder vielleicht sogar einem anderen Team-Mitglied den Fall überlassen. Auch innerhalb des Unternehmens sind einige Gratwanderungen zu meistern. Ein Community Manager braucht Informationen und sollte sie bekommen. Dazu gehört das Vertrauen, dass er mit den Informationen so umgeht, wie man ihn bittet, es zu tun. Er darf dieses Vertrauen nicht enttäuschen!

Arbeitet er für ein Unternehmen, das seine Arbeit auch in der Form würdigt, dass seine Vorschläge, Warnungen und Rückmeldungen ernst genommen und umgesetzt werden, darf er dennoch nicht hingehen und die vorhandenen Strukturen ausnutzen oder ausreizen. Auch er muss die oben genannten Schnittstellen respektieren und mit ihnen zusammenarbeiten.

Wo ist die Community?

Wir haben bereits angeschnitten, dass es eine Unterscheidung zwischen den sichtbaren und den unsichtbaren Community Mitgliedern gibt. Gehen wir von dem Beispiel eines offiziellen Forums aus, so kann man sagen, dass nur etwa fünf bis zehn Prozent der Gesamtcommunity dort aktiv und somit sichtbar sind. Die übrigen lesen nur mit oder sind in Subcommunitys zu finden, wie zum Beispiel in Gilden oder auf den bereits erwähnten Fanseiten.

Eine wichtige Regel für ein gutes und erfolgreiches Community Management lautet, so viele Community Mitglieder wie möglich zu erreichen. Dazu müssen zwei Dinge beachtet werden: Das Community Management muss dahin gehen, wo die Community ist. Wo ist die Community? Im Spiel, natürlich. Trotzdem liegt der grundsätzliche Wirkungsbereich eines Community Managers nicht im Spiel, sondern außerhalb. Sich dabei nur auf das offizielle Forum zu fokussieren, bedeutet das Verschwenden von wichtigen Ressourcen aus der Community. Man muss auch mit den Gilden und den Fanseiten reden, Interviews in Printmedien geben – abseits von Marketing-Aktionen – und sich auf Veranstaltungen, wie zum Beispiel Messen oder vielleicht auch Gildentreffen, sehen lassen.

Wir wissen nun also, wo ein Community Manager kommunizieren sollte, aber immer noch nicht, wie er dies tun sollte, um möglichst viele Community Mitglieder zu erreichen und vor allem auch den stummen, unsichtbaren Mitgliedern Gehör zu verschaffen. Viele Community Manger werden gerade zu Beginn in die gleiche Falle tappen: Sie werden denjenigen oder die Gruppe hören, die am lautesten ruft. Aber ist diese repräsentativ? Manchmal, aber bei Weitem nicht immer. Diese Gruppen zu ignorieren, wäre jedoch genauso falsch.

Der Community Manager muss die verschiedenen Mitglieder einer Community identifizieren können. Wer ist ein Leitwolf und warum? Weil er laut brüllt oder weil er mit seiner konstruktiven Kritik vielen der etwas Stilleren oder ganz Unsichtbaren aus dem Herzen spricht? Wer will einfach nur Stress verbreiten, wer ist wirklich wütend? Wer ist vielleicht gar viral unterwegs? Welche Community Mitglieder sollte man ermutigen, weil dadurch ein Schneeballsystem in Gang gebracht werden kann? Welche Kritiker könnte man dazu bringen, so konstruktiv mitzuarbeiten, dass sie zu wertvollen Sprachrohren werden?

Der ideale CM

Welche Fähigkeiten soll der Community Manager mitbringen? Viele: Er muss mit Leidenschaft an eine Sache herangehen können und doch auch kühl kalkulierte Fakten verstehen und nachvollziehen. Er muss sensibel und für Stimmungen empfänglich sein und braucht doch ein dickes Fell. Er muss Spaß an einem Spiel haben können und doch die Vorgänge verstehen, die zu der Entwicklung und dem Betrieb eines Spiels gehören. Er muss sich sehr gut ausdrücken können und darf doch nicht zu konstruiert klingen. Er muss flexibel in vielen Bereichen sein und braucht doch feste Eckpfeiler, an die er sich halten kann. Er muss ein Entertainer sein können und darf sich doch nicht zum Affen machen. Er muss Kumpel sein und gleichzeitig Respektperson. Er muss ein Einzelkämpfer sein, der zu 100% teamfähig ist.

Er ist ein Sprachrohr in zwei Richtungen, quasi ein doppelter Simultanübersetzer: vom Unternehmen zur Community und von der Community zum Unternehmen. Idealerweise versteht er nicht nur die Sprache der Community, sondern auch die der Entwickler, der Kaufleute, des Marketings, der verschiedenen Kultur- und Sprachkreise und so weiter und so fort. Ein Community Manager zerreißt sich quasi ständig selbst. Es kommt nicht von ungefähr, dass man sagt, dass ein Community Manager eine »Halbwertzeit« von circa drei bis fünf Jahren hat. Danach ist er ausgebrannt, hat sich verausgabt, da er ständig am Limit steht. Auch hier daher die Empfehlung: Lassen Sie ihn nicht alleine kämpfen!

Wenn ein Team für Sie zum Beispiel aus Gründen des Budgets oder der Unternehmensstruktur nicht in Frage kommt, dann sollten Sie zumindest hingehen und Folgendes tun: Gewähren Sie ihrem Community Manager einen ausführlichen Einblick in alle relevanten Abteilungen, bevor er die Arbeit aufnimmt – und stehen Sie ihm auch danach so oft wie möglich mit angefragten Informationen und Antworten zur Seite.

Die Community ist das Herz eines Spiels – und das Community Management ist die Lunge. Das Herz schlägt nicht immer im Takt, aber die Lunge sollte immer für ausreichend Luft sorgen.

Nadja Bastawi, Thomas Lagemann

World of Warcraft: »Für ihren Support mussten Blizzard Kritik einstecken, dennoch machen sie auch vieles richtig.«

»It's not a game, it's a service!«

In-Game-Support in MMOs

Game Master sollen die Probleme der Fans direkt in Online-Spielen lösen. Teut Weidemann erklärt, welche Werkzeuge hierfür notwenig sind – und welche Grundregeln niemals gebrochen werden dürfen.

Teut Weidemann

arbeitet als Unternehmensberater.

Mit 22 Jahren Branchenerfahrung ist Teut Weidemann einer der kundigsten Spieleproduzenten Deutschlands. 1987 stieg er als Development Director bei Softgold und Rainbow Arts ein, dem damals größten deutschen Publisher. 1993 wurde er Entwicklungsleiter bei Rauser Advertainment, dem Erfinder von Werbespielen.1996 gründete er das Studio Wings Simulations, mit dem er 1999 die Simulation Panzer Elite und 2004 den Mehrspieler-Shooter Söldner: Secret Wars veröffentlichte, der sich über 350.000 Mal verkaufte. 2005 wechselte Weidemann als CTO zu CDV. Heute greift er Unternehmen als selbstständiger Berater unter die Arme.
www.teut.blogspot.com

Derzeit gelten die sogenannten MMOGs (Massive Multiplayer Online Games, auch »MMOs«) als Wachstumsmotor der PC-Spielebranche. Doch viele Hersteller stürzen sich auf das Thema, ohne den größten Unterschied zwischen den traditionellen PC-Spielen und den MMOs zu erkennen: die Community! Raph Koster, der Lead Designer von Ultima Online, sagte bereits in den 90er Jahren über das MMO: »It's not a game, it is a service« – zu deutsch: »Es ist kein Spiel, sondern eine Dienstleistung.« Aber was heißt das überhaupt? Und warum ist dieses Motto so wichtig, dass es einige Firmen zum unumstößlichen Grundsatz erheben?

Die Spielmeister

Ich konzentriere mich in diesem Artikel auf den In-Game-Support, der von so genannten Game Mastern (GMs) geleistet wird. Zunächst müssen wir aber zwischen drei Begriffen unterscheiden: GMs, Community Managern (CMs) und Moderatoren. GMs leisten ihren Support »one on one«. Das heißt, dass sie dem Kunden in einem privaten Gespräch Hilfestellung geben – für gewöhnlich direkt im Spiel. Community Manager hingegen veröffentlichen ihre Nachrichten und Tipps vorrangig in Foren. So sprechen sie alle Spieler an, die dort mitlesen. Gleiches gilt für die Moderatoren. Da sich die Arbeitsweise eines GM und eines CM komplett unterscheiden, gilt die Grundregel: GMs haben im Forum nichts verloren! Niemals! Punkt! Umgekehrt müssen CMs und Moderatoren die Finger von GM-Tätigkeiten lassen – sie dürfen sich also nicht in Privatgesprächen an einzelne Kunden wenden. Falls eine Firma glaubt, dass all diese Jobs von ein und derselben Person erledigt werden könnten, legt sie den Grundstein für heilloses Chaos. Der Hauptgrund liegt in der Art und Weise, wie man mit dem Spieler umgehen muss und wie viel Spielraum man dabei hat. Der CM genießt nämlich sehr viel mehr Freiheiten als der GM.

Ohne Namen

Ein GM ist immer anonym und tritt mit seinem Nicknamen auf, also seinem Spitznamen im Spiel. Außerdem ist ein GM immer formell. Das heißt, er spricht über das Problem des Spielers – und sonst über gar nichts. Auf keinen Fall darf er persönliche Themen in den Support einbringen, seinen echten Namen preisgeben oder sich bei Spielerfreunden als GM »outen«. Das hat auch historische Gründe: In Ultima Online war es ganz normal, dass GMs im Spiel sowie im Forum und in Chats unterwegs waren. Dort wurde dann über alles Mögliche gesprochen, schnell sickerten die echten Namen einiger Game Master durch. Das war auch kein Problem, bis einem

Spieler der Account gesperrt wurde. Daraufhin lauerte der wütende Betroffene dem schuldigen GM auf – und verprügelte ihn!

Alle Spieler sind gleich

»Jeder Kunde muss gleich behandelt werden!«, so lautet das wichtigste Gesetz der GMs. Ich erkläre es anhand eines Beispiels aus einem MMO: Ein Spieler hatte einen mächtigen Gegenstand von einem Gegner erbeutet – doch aus irgendwelchen Gründen fand er diesen Gegenstand nicht in seinem Inventar. Die Beute war einfach verschwunden. Der GM gab dem Abenteurer den Gegenstand zurück, da scheinbar ein technischer Fehler vorlag. Doch der Spieler erzählte im Forum davon, und prompt meldeten sich Dutzende weiterer Fans, die ebenfalls einen Gegenstand vermissten. Der GM war allerdings nicht bereit, alle verlorenen Beutestücke zu ersetzen. Das führte zu Unruhe, einige Spieler kündigten ihren Account. Andere hatten fortan keine Lust mehr, einen GM zu Hilfe zu rufen. Die Ungleichbehandlung hatte also schwerwiegende Folgen und wühlte die Community auf.

Und wie muss man vorgehen, wenn man jeden Kunden gleich behandeln möchte? Zunächst muss der Hersteller verbindliche Regeln für den Umgang mit Kunden aufstellen. In World of Warcraft etwa führte das Gesetz der Gleichbehandlung so weit, dass selbst der exakte Wortlaut der Problemlösungen vorgegeben war. Die GMs antworteten einfach mit vorformulierten Text-Makros. Dies änderte sich erst, als sich die Spieler fortwährend über den GM-Support beschwerten. Daraufhin ließ Blizzard seinen GMs etwas mehr Freiheiten – allerdings nur in einem bestimmten Rahmen.

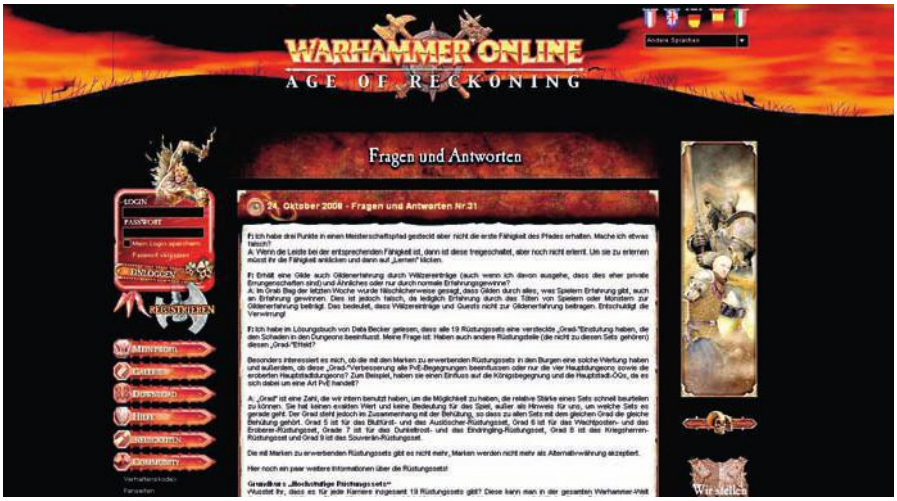
Die Kundenverwaltung

Traurig, aber wahr: Manche Spieler nutzen Schwachstellen im Programmcode oder im Support aus, um sich unfaire Vorteile zu verschaffen. Der GM muss daher sicherstellen, dass ein Kunde bei seiner Beschwerde die Wahrheit sagt. Um bei obigem Beispiel zu bleiben: Es muss ein Werkzeug geben, mit dem der GM prüfen kann, ob der Spieler den vermissten Gegenstand wirklich in der Tasche hatte.

Dieses Prüfwerkzeug ist das CMS-Tool. Die Abkürzung steht für »Customer Management System«, zu Deutsch: »Kunden-Verwaltungssystem«. Solche Programme werden von vielen Firmen eingesetzt, Versicherungen etwa archivieren ihre Kundenvorgänge in CMS-Datenbanken. Betreiber großer MMOs lizenzieren oft für viel Geld eine CMS-Software und passen diese danach an ihr Spiel an. Kleine Hersteller entwickeln ihr CMS-Tool selbst oder nutzen eine der zahlreichen Open-Source-Lösungen. Doch was muss ein solches Werkzeug überhaupt leisten?

Ticket für Hilfe

Jeder Spieler, der einen GM zu Hilfe ruft, reicht ein sogenanntes »Support Ticket« ein. Da die GMs alle Server eines Spiels betreuen, müssen



Support-Website von Warhammer Online: »Der GM kann bei Anfragen einfach auf diese Internet-Adresse verweisen.«

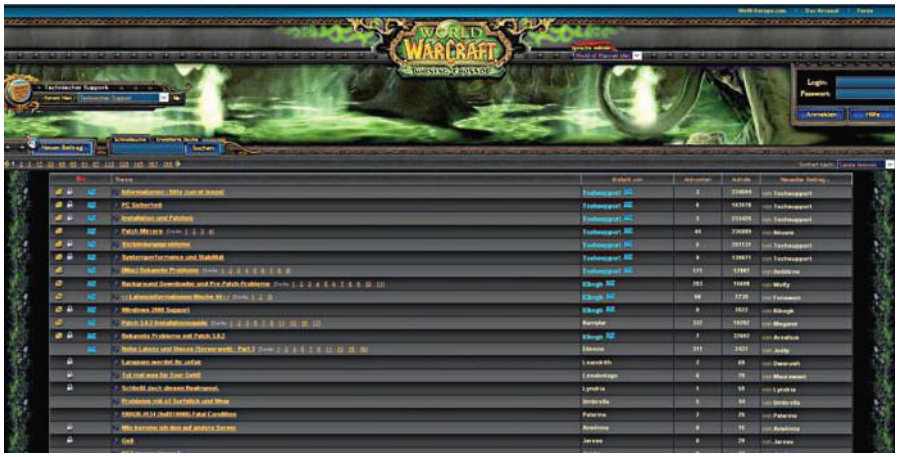
sich diese Anfragen im CMS-Tool zentral sammeln, organisieren, bearbeiten und nach ihrer Priorität sortieren lassen. Einige Firmen arbeiten zudem mit einem 24-Stunden-Support, dessen Mitarbeiter meist über mehrere Zeitzonen verstreut sind. Das Ticket-System muss also auch mehrere Standorte verbinden können. Das Gesetz zur Gleichbehandlung verlangt überdies, dass die Antwortzeiten der GMs weltweit ungefähr gleich sein sollten. Ein durchdachtes Ticket-System kann dies gewährleisten.

Wer ist das?

Ein GM, der im Spiel mit einem Kunden spricht, muss jederzeit weitere Informationen über seinen Gesprächspartner abrufen können. Unter anderem sollte er sehen, wie lange der Kunde bereits spielt und welche Support-Anfragen er schon gestellt hat. So vermeidet man Missverständnisse sowie peinliche Situationen à la »Der andere GM hat aber etwas ganz anderes erzählt!«. Ebenfalls wichtig ist, ob der Kunde von den CMs im Forum verwarnt oder gesperrt wurde. So lassen sich chronische Störenfriede identifizieren. Normalerweise stehen solche Zusatzinfos nicht direkt im speziellen Client des GM, sondern in einem gesonderten Programm, welches mit dem GM-Client kommuniziert – eben dem CMS-Tool. Doch wie kann ein GM damit sehen, ob ein Spieler einen bestimmten Gegenstand verloren hat?

Stasi lässt grüßen

Der Stasi-Vergleich klingt zwar hart, aber er umreißt die wichtigste Regel für guten Support: Ein MMO mit Millionen Spielern muss alles protokollieren (»loggen«): Chats, erbeutete Ausrüstung, Handelsgeschäfte mit Gegenständen und Währung, erledigte Monster sowie Bosskämpfe. Der Entwicklungs- und Kostenaufwand für dieses so genannte »Tracking« ist enorm und wird vor allem von unerfahrenen Entwicklern und Publishern gescheut. Doch dabei übersehen diese Firmen, dass auch mangelndes Tracking enorm viel Geld kosten kann.

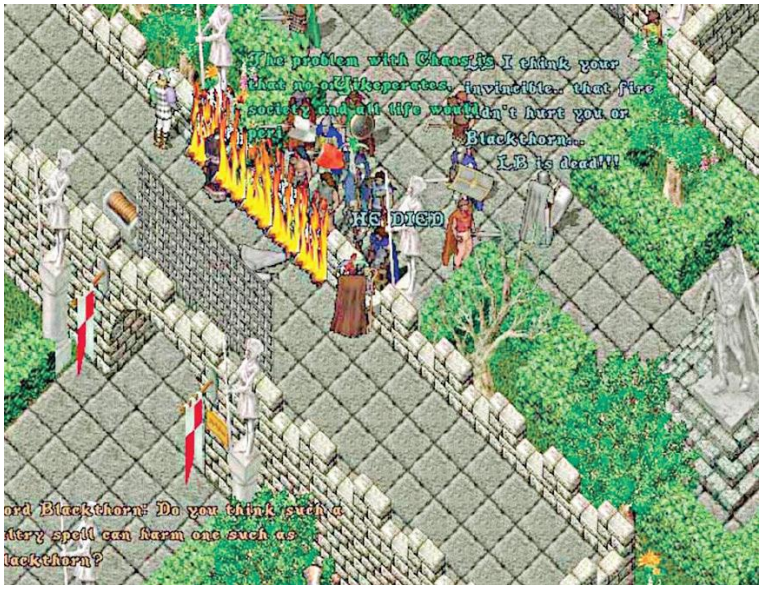


Offizielles Forum von World of Warcraft: »GMs haben im Forum nichts verloren! Niemals! Punkt!«

Ein Beispiel aus Ultima Online: Die minderjährige Tochter eines Spielers steuerte gerne dessen Charaktere, darunter eine hübsche Elfe. Dabei wurde sie im Chat mit sexuellen Andeutungen konfrontiert, die sie selbst nicht wirklich verstand. Als danach der Vater die Elfe spielte und eindeutig zweideutige Nachrichten bekam, war er entsetzt – und verklagte den Betreiber des Spiels kurzerhand wegen sexueller Belästigung Minderjähriger. Weil die Firma kein Chat-Protokoll vorweisen konnte, um die beteiligten Personen zu identifizieren, endete der Prozess in einem Vergleich. Seit diesem Vorfall archivieren vor allem amerikanische MMOs alle Chat-Botschaften so lange, bis die Klagefristen verstrichen sind. Da diese für jedes Land unterschiedlich sind, bedeutet das Gesprächs-Tracking einen erheblichen Aufwand – zumal sich bei großen Titeln weltweit Millionen Spieler unterhalten.

Verfolgte Millionen

Doch auch abseits der Gespräche müssen die Protokolle sehr genau sein. In World of Warcraft etwa kann ein GM genau sehen, wann, wo und wie der Spieler einen Gegenstand gefunden hat, und wer zu diesem Zeitpunkt in seiner Gruppe war. Zudem kann der GM feststellen, ob der Gegenstand weitergegeben, zerstört oder im Auktionshaus verkauft wurde. Dies soll verhindern, dass Spieler sich mit Hilfe der GMs Gegenstände erschleichen. Dies klappt aber nur, wenn jeder Gegenstand im Spiel eine einzigartige Identifikationsnummer (ID) besitzt, anhand derer man ihn verfolgen kann. Bei Millionen Gegenständen ist dies eine knifflige Aufgabe.


The problem with Chaos is I think your
that no one's operates... invincible... that fire
society and all life wouldn't hurt you or
per!
Blackthorn...
L.B is dead!!!
HS DISP
ord Blackthorn: Do you think such a
tery spell can harm one such as
lackthorn?

Ultima Online: »Ein Vater verklagte den Betreiber wegen sexueller Belästigung Minderjähriger.«

Gegen den Missbrauch

Wie bereits erwähnt, kann Missbrauch zum zentralen Problem eines MMOs avancieren. Denn manche Spieler denken sich nichts dabei, wenn sie die Spielregeln brechen und Schwachstellen nutzen, um sich Vorteile zu verschaffen. Es ist halt »nur ein Spiel«. Dabei verkennen die Nutznießer jedoch, dass sie dem MMO schaden. Wenn sich ein Spieler etwa Gegenstände erschleicht und verkauft, kann er den Währungs- und Wirtschaftskreislauf ruinieren. Und wer seinen Charakter mit erschummelten Fähigkeiten zum Superkrieger ausbaut, verdirbt anderen Abenteurern schlichtweg den Spaß. Hier müssen die GMs einschreiten.

Manche Missbrauchsfälle lassen sich allerdings auch von vorneherein verhindern, indem man simple Techniklücken schließt. So ist es in den meisten MMOs üblich, dass ein GM seine Chat-Botschaften in einer exklusiven Farbe schreibt. Gewiefte Spieler finden häufig eine Möglichkeit, diese Farbe zu kopieren und sich als GM auszugeben. Daher ist es notwendig, dass der normale Client diese Farbe gar nicht erzeugen kann. Zudem muss die Farbe einer Chat-Botschaft stets vom Server geprüft und verifiziert werden. Das gilt auch für alle anderen wichtigen Systemmeldungen, etwa Würfelergebnisse.

Kontrollierte Kontrolleure

Die Paranoia lässt sich allerdings auch umdrehen: Wer kontrolliert überhaupt die Game Master? Wenn ein GM Gegenstände erzeugen und verteilen kann – wer verhindert dann, dass er seine Freunde mit den tollsten Gaben beschenkt? Der Support eines erfolgreichen MMOs kann gewaltige Ausmaße annehmen, bei Blizzard etwa arbeiten daran alleine in Europa 500 Personen. Also ist die Kontrolle der Helfer ebenfalls ein wesentlicher Faktor.

Denn Vertrauen alleine reicht leider nicht, wie ein weiteres Beispiel aus Ultima Online zeigt: Ein GM erschummelte sich Gegenstände, um Schlösser zu bauen – die teuersten und größten Häuser. Zudem pflanzte der GM die Gebäude in Gegenden, in denen normale Spieler gar keine Baugenehmigung bekamen. Seine Häuser versteigerte er dann für Tausende Dollar im Internet. Als der Hersteller dies erfuhr, feuerte er den GM. Doch zugleich verlor die Community für geraume Zeit das Vertrauen in die Firma und die GMs – ein riesiger Image-Schaden.

Immer überwachen

Die Aktionen eines GMs müssen also ebenfalls protokolliert werden. Einige Optionen sollten zudem den Lead-GMs vorbehalten bleiben – insbesondere Eingriffe, die große Auswirkungen auf die Spielwelt haben. Es ist schließlich schwer, alle Support-Mitarbeiter ständig zu kontrollieren. GMs sollten daher die Möglichkeit haben, ein haariges Kundenproblem an einen leitenden Mitarbeiter weiterzureichen. Es ist wichtig, dass der Client der GMs diese Hierarchie so fest verdrahtet, dass Missbrauch grundsätzlich ausgeschlossen ist.

Und wie weit muss die Kontrolle der GMs gehen? Sehr weit. So weit, dass selbst ihre E-Mails, Telefonate sowie Instant-Messenger-Nachrichten überwacht werden. Denn die GMs sitzen direkt an der Quelle; ihr Arbeitgeber muss sicherstellen, dass sie niemals vertrauliche Informationen an Dritte weitergeben. Vor allem dann nicht, wenn diese Informationen später exklusiv der Community enthüllt werden sollen. Denn der Geheimnisverrat würde den Informationsfluss an die Fans und damit das Wachstum der Kundenbasis stören: Warum sollten die Spieler auf den Hersteller hören, wenn sie ihre Infos auch anderswo bekommen?

Ein Beispiel: Auf einer deutschen Fanseite tauchten vorab Infos zum nächsten Patch eines bekannten MMOs auf. Es dauerte nur Minuten, bis diese ins Englische übersetzt und im gesamten Internet zu lesen waren. Als der Hersteller die E-Mails seiner Support-Mitarbeiter untersuchte, war der Schuldige schnell gefunden: Ein GM hatte die Update-Details »nur einem guten Freund zeigen wollen«. Die Reaktionszeit zwischen dem Geheimnisverrat und dem Sperren des Mitarbeiters dauerte dank des überwachten Mail-Verkehrs nur Stunden, nicht Tage. Trotzdem wurden das Marketing und die PR für diesen Patch erheblich gestört.

Die Problembibel

Im Regelfall kennen die GMs die meisten Probleme ihres Spiels, schließlich sitzen sie direkt an der Quelle, also beim Hersteller. Überdies hören sie oft viele Beschwerden zum gleichen Thema. Denn auch wenn der Kunde selbst dies nicht wissen mag, ist er meistens nicht der Erste, der sich mit einer Anfrage an einen GM wendet. Daher führen die Betreiber aller größeren MMOs eine Datenbank, in der alle Anfragen gespeichert werden (eine sogenannte »Knowledge Base«) – und die normalerweise rund 90 Prozent aller Probleme abdeckt.

Diese Datenbank ist auch die Voraussetzung dafür, dass alle Kunden gleich behandelt werden. Denn sie listet bei jedem Problem auf, wie sich der GM zu verhalten hat, welche Lösungsvorschläge er geben kann und welche Position er einnehmen soll. Im Idealfall enthält die Knowledge Base sogar Textbausteine, die der GM einfach per Mausklick in seine Chat-Botschaften übernehmen kann. Das bringt Effizienz, denn bei Millionen Spielern muss ein GM schnell vielen Leuten helfen können. Zudem sollte der Hersteller die häufigsten Probleme samt Lösung auf einer Support-Webseite sammeln. Dann kann der GM bei Anfragen einfach auf diese Internet-Adresse verweisen.

Der Mensch GM

Dass sich ein GM gut im Spiel auskennen sollte, versteht sich von selbst. Deshalb werden die Helfer meist aus der Community rekrutiert. Doch weil der GM das MMO auch weiterhin privat spielen wird, muss er anonym bleiben. Ihm muss klar sein, dass er gegenüber seinen Gildenkollegen und Freunden im Spiel niemals bekannt geben darf, dass er ein GM ist.

Die Betreiber des MMO sollten überdies beachten, dass jeder GM eine bestimmte »Lebensdauer« besitzt. Er steht morgens auf und steigt als GM ins Spiel ein, dann spielt er bis abends privat weiter. Innerhalb der Firma begegnet er ausschließlich anderen Support-Mitarbeitern, mit denen er meist nur über ein Thema spricht: das Spiel. Kurz gesagt: Das Leben des GM kreist um ein einziges Programm. Deshalb brennen die meisten GMs irgendwann aus und hören auf. Der MMO-Betreiber muss also fortwährend Nachwuchs suchen und ein Ausbildungsprogramm aufziehen, um Neulinge rasch einzuarbeiten. Vor allem sollte die Firma ihre Mitarbeiter beim Umzug unterstützen, denn die Community-Mitglieder sind meist über das ganze Land verstreut. Oder sogar über die ganze Welt: Da ein MMO nur dann die kritische (lies: lohnende) Spielerzahl erreicht, wenn es weltweit online geht, sind GMs meist mehrsprachig oder stammen sogar aus anderen Nationen. Die Integration in die Firma und ins fremde Land fällt ihnen entsprechend schwer und beschleunigt das oben beschriebene Burn-Out-Syndrom.

Lohnender Aufwand

Für ihren Support mussten Blizzard zwar viel Kritik einstecken, dennoch machen sie auch vieles richtig. Denn für Blizzard stand stets die Community im Vordergrund. Nicht umsonst kündigen die Kalifornier ihre neuen Spiele auf Fan-Veranstaltungen an – und nicht auf Messen oder bei PR-Terminen.

Doch lohnt sich ein aufwändiger Support überhaupt? Muss das sein? Ich drehe die Frage mal um: Ist dieser Aufwand nicht Voraussetzung für ein erfolgreiches MMO? Denn das Wachstum der Spielerbasis wird selten durch Marketing und PR gefördert, es geschieht größtenteils innerhalb der Community durch Weiterempfehlung!

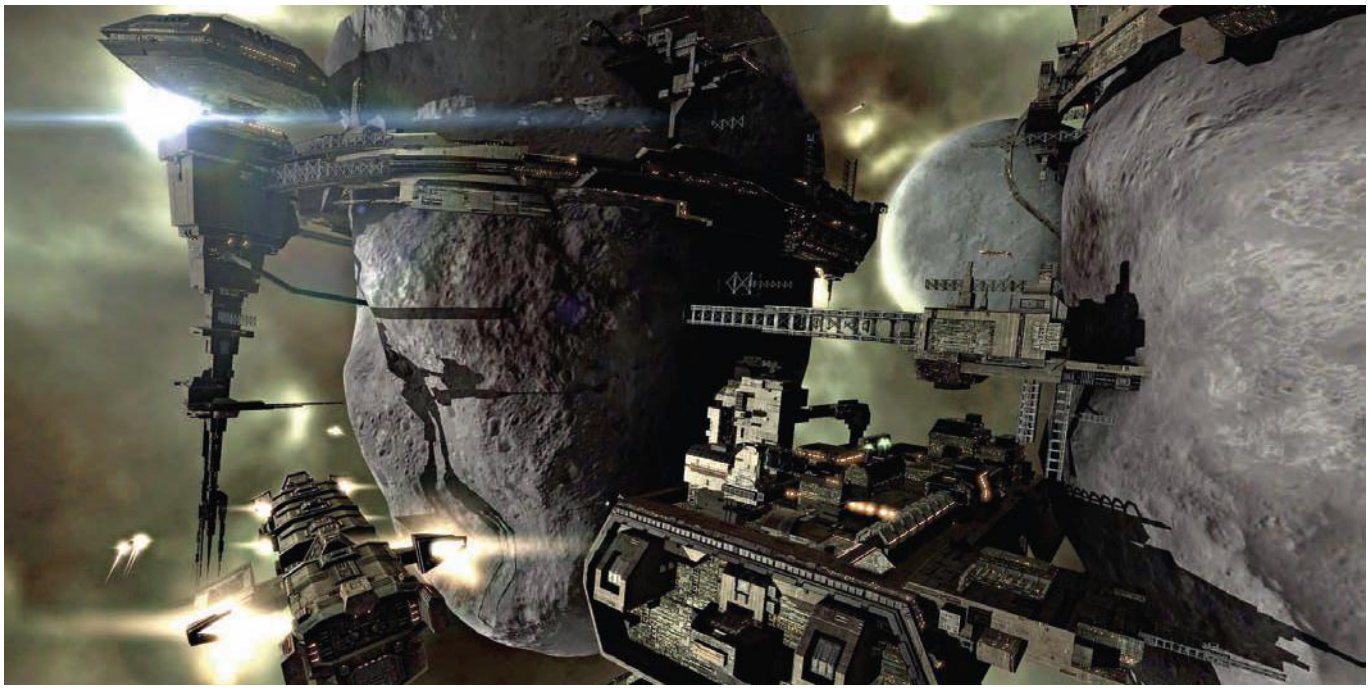
Teut Weidemann

Wichtige Begriffe

- **Avatar:** Die Figur, die der Spieler steuert und mit der er sich in der virtuellen Welt präsentiert.
- **CM:** Community Manager, eine Person, die im Support arbeitet und dabei öffentlich auftritt, meist in den Foren oder auf der Webseite.
- **CMS:** Customer Management System, eine Software, die den Support organisiert. Weil sie Anfragen sammelt, ermöglicht sie, dass mehrere Support-Mitarbeiter denselben Kunden betreuen.
- **Community:** Die Menschen, die Ihre Welt spielen oder Ihre Webseite besuchen. Derzeit ein Inbegriff des Missverstehens, vor allem bei Investoren, bei CEOs und beim Marketing.
- **GM:** Game Master, eine Person, die im Support arbeitet und innerhalb des Spieles agiert.
- **God Client:** Eine spezielle Variante des Spiels, die einem GM erlaubt, fast alles zu verändern. Zum Beispiel die Gegenstände oder die Eigenschaften eines Charakters. Der Betreiber eines MMOs muss unbedingt verhindern, dass ein God Client außerhalb seines Netzwerkes läuft.
- **MMO(G):** Massive Multiplayer Online (Game), ein Programm, in dem mehrere Tausend Menschen in einer zusammenhängenden Spielwelt mit- oder gegeneinander spielen. Weil die Software bestimmte Grenzen hat, sind die Welten meist in Spiegelwelten unterteilt.
- **MMORPG:** Massively Multiplayer Online Role Playing Game, Erweiterung des Begriffes MMO, auf Rollenspiele beschränkt. Die meisten erfolgreichen MMOs sind Rollenspiele, weil sie die Spieler durch Belohnungen wie neue Waffen oder Fähigkeiten langfristig binden.
- **Tickets:** Ein System im Spiel, mit dem der Kunde abrufen kann, wie weit sein Support-Anruf (also sein »Ticket«) in der Bearbeitung ist. Außerdem erleichtert es dem GM, die Anfragen nach Priorität, Sprache und Server zu bearbeiten.
- **Ultima Online:** Eines der ersten wirtschaftlich erfolgreichen MMORPGs, erschienen 1997. Mit über 300.000 Spielern war es eine der größten westlichen Online-Welten, die heute immer noch rund 100.000 Spieler bindet.

User-Mitbestimmung in Eve Online
Eine symbiotische Beziehung

Die Spielergemeinschaft von Eve Online gestaltet das virtuelle Leben im Online-Universum zu einem großen Teil selbst – seit einiger Zeit sogar mit einem Rat aus demokratisch gewählten Vertretern.



Eve Online ist eine **»Single Shard World«**, an der weltweit alle Spieler gemeinsam teilnehmen, ohne sich auf unterschiedliche Server aufzuteilen.

Dr. Eyjólfur Guðmundsson

ist Lead Economist bei Eve Online.

Dr. Eyjólfur Guðmundsson hat an der Universität von Rhode Island (USA) Environmental and Resource Economics (Ph.D) studiert. Bevor er 2007 zu CCP wechselte, war er Dekan für die Fakultät Business and Science an der Universität von Akureyri (Island). Bei CCP verantwortet er die Überwachung und Analyse der Ökonomie von Eve Online.

Die Community der Spieler von Eve Online stand in CCPs Gedanken schon immer an erster Stelle, sogar bevor die virtuelle Welt von New Eden vor über fünf Jahren online ging. Mit den größten Hoffnungen begannen wir, ein Online-Universum zu schaffen, an dem alle Spieler weltweit gemeinsam teilnehmen sollten, ohne sich für einen bestimmten Server entscheiden zu müssen (»Single Shard World«). Wir wollten eine virtuelle »Sandbox« schaffen, die den Teilnehmern alle Möglichkeiten gibt, nach ihren eigenen Moralvorstellungen zu handeln und ihre eigenen Kriege zu führen. Seit Anbeginn der Konzeption von Eve wussten wir, dass der virtuelle Tod des Spielers in Eve nicht ohne Konsequenzen sein durfte, wenn wir unsere Ziele erreichen wollten. Wenn Ihr Schiff explodiert, erscheint es nicht auf magische Weise nur leicht beschädigt in der nächsten Raumstation. Stattdessen werden Sie mit Ihrer Rettungskapsel aus den Überresten Ihres einst stolzen Schiffes katapultiert – meist noch in Sichtweite derjenigen, die Ihren wertvollen Besitz nur Momente zuvor zerstört haben. Ihr Schiff und die Millionen oder gar Milliarden, die Sie dafür ausgegeben haben sowie die für Zusammenbau und Ausrüsten aufgewendete Zeit sind verloren. Selbst eine Versicherung (falls abgeschlossen) gleicht diesen Verlust nur zum Teil aus.

Kein Tod ohne Risiko

Es ist das drohende Gespenst des Verlustes, das die Spieler jedes Mal begleitet, wenn sie von einer Station abdocken. Dieses stete Risiko verleiht den sozialen Interaktionen zwischen den Spielern wesentlich mehr Bedeutung als in anderen Online-Spielen. Denn Vertrauen ist somit das wertvollste Gut, dass Sie jemandem schenken können – gleichzeitig hat es die verheerendsten Auswirkungen, falls jemand Ihr Ver-

trauen missbraucht. Wenn Sie die Arbeit von Monaten in nur wenigen Minuten verlieren können, werden Ihre Kameraden unendlich viel wichtiger als die Ausrüstung Ihres Schiffes. Deshalb schließen sich unsere Spieler wie im echten Leben zusammen – für mehr Sicherheit und Profit, für Kameradschaft und Krieg. Dabei bieten hoch begehrte Ressourcen im tiefen Weltraum, wo keine schützende Polizei patrouilliert, den unwiderstehlichen Anreiz, diese Gebiete für sich zu beanspruchen, neue Grenzen zu schaffen und mit Verbündeten zu verhandeln.

Die Spieler entscheiden selbst

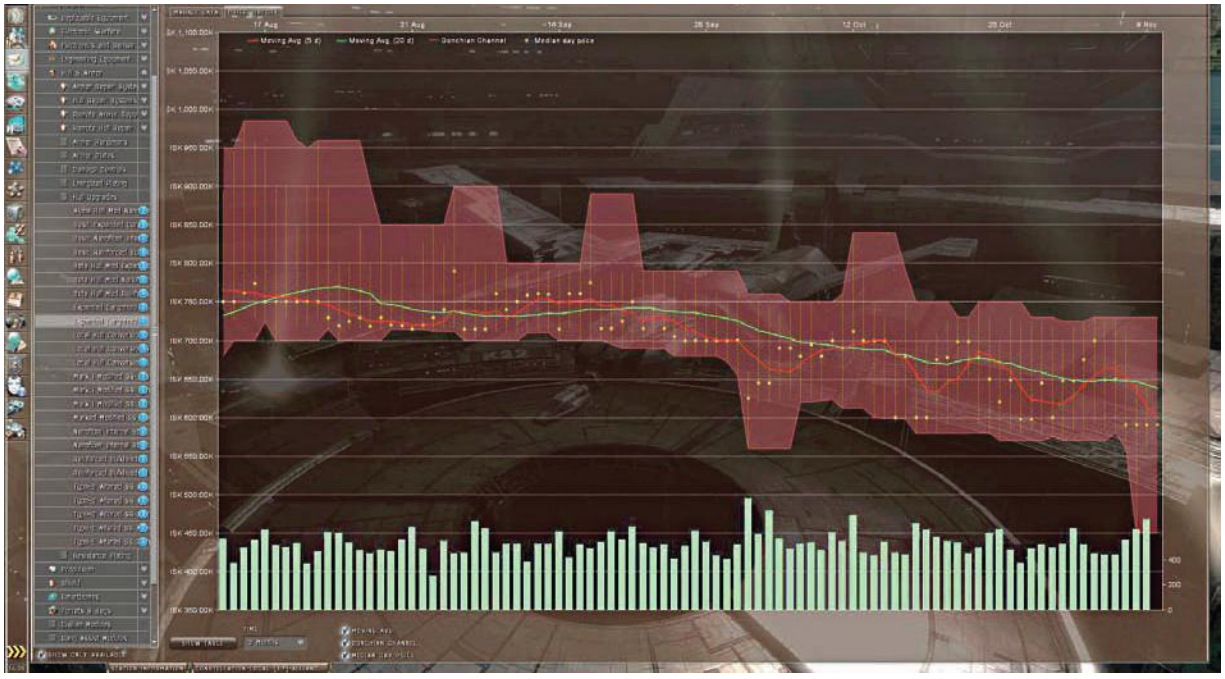
Aus oben genannten Gründen halten wir uns bei der Einflussnahme auf die Spielergemeinschaft von Eve eher zurück. Wir stellen die Strukturen, die die Spieler für die Nachschubsproduktion, für ihre Corporations (Spielergemeinschaft in Eve) oder den Handel nutzen. Dinge wie das Vertragssystem, der Marktplatz, das Hoheitssystem und die Rechteverwaltung in einer Corporation bieten engagierten Spielern die Möglichkeit, selbst zu entscheiden, bis zu welchem Grad sie an Eve teilnehmen. Aber eines der wesentlichen Merkmale ist, dass man auch ohne diese ganzen Systeme spielen kann. Sie werden nicht gezwungen, einer Spieler-Corporation beizutreten. Sie müssen Ihre Deals nicht über das Vertragssystem abwickeln. Nicht einmal die Teilnahme am regionalen Markt ist notwendig, solange Sie gut im Feilschen, Produzieren oder gar im Stehlen sind.

Den Spielern folgen

Da Eve eine Single Shard World ist, hat die stetig steigende Gesamtzahl der Spieler beim Überschreiten gewisser Grenzen tiefgreifende Veränderungen in der Community und auch bei CCP bewirkt. Diese aufgrund der wachsenden Spielerzahl eintretenden Veränderungen nennt man »macro-emergent behaviour«. Darunter versteht man die Verhaltensänderungen, die man in der Masse der Spieler erkennt, wenn man diese von der übergeordneten Makro-Ebene aus betrachtet.
Einer dieser Momente war zum Beispiel der Aufstieg der Allianzen in Eve. Anfangs konnte das vorhandene Corporation-System genug Piloten für die meisten Operationen in New Eden zusammenbringen. Als immer mehr Spieler Eve beitraten, begannen ähnlich gesinnte Corporations erste, informelle Allianzen zu formen. CCP hatte dieses Verhalten auf der konzeptionellen Ebene bereits vorhergesehen. Trotzdem haben wir das Allianzsystem nicht von Entwicklerseite aus eingeführt , sondern gewartet, bis es auf natürliche Art bei den Spielern Einzug hielt und erst dann die nötigen Strukturen ins Spiel eingebaut. Das war ein Paradebeispiel, wie unsere Spieler sich anpassen und das Gameplay in New Eden vorantreiben.

Eine weitere Veränderung war, als wir dazu übergingen, den Markt nicht mehr künstlich mit Waren zu versorgen, sondern zu reinen Beobachtern wurden. Am Anfang von Eve musste CCP mit Käufen und Verkäufen aktiv am Markt teilnehmen, um sicherzustellen, dass genug Material und Gegenstände im Handel waren. Als die Markteilnehmer mit wachsender Spielerzahl stiegen, konnten wir die Aufkäufe immer mehr reduzieren. Als die Spielerzahl die 50.000 überschritten hatte, konnten wir den Markt schließlich sich selbst überlassen und uns als Hauptakteur zurückziehen. Heute werden die gesamten Waren in über 60 regionalen Märkten von den Spielern alleine erzeugt, transportiert und verkauft. Mit der Ausnahme einiger Skill-Bücher (zum Erlernen gewisser Fähigkeiten) wird alles von den etwa 250.000 Spielern selbst hergestellt. Es ist eine echte selbstständige Wirtschaft, die allen theoretischen ökonomischen Definitionen gerecht wird. Sie spiegelt in vielen Dingen das typische Marktverhalten wider, wie man es aus dem echten Leben kennt. In den größeren Handelsplätzen sehen wir die Bedingungen eines perfekten Marktes fast erfüllt, wo kein Spieler den Markt zu seinen Gunsten beeinflussen kann. In bevölkerungsärmeren Regionen können und werden die Märkte dagegen mit Hilfe von Angebot und Nachfrage manipuliert – etwa von findigen Geschäftsleuten oder als Teil der Kriegsstrategie einer Allianz in der Region. Wie im echten Leben ist Krieg eine bemerkenswert starke Marktkraft. In New Eden toben jeden Tag massive Konflikte, bei denen Spieler Werte verlieren oder gewinnen, in dem sie die Wracks ihrer Gegner plündern. Wer verliert, muss sich zunächst wieder der Produktion neuer Schiffe und Munition für die nächste große Flottenschlacht oder das nächste kleine Scharmützel zuwenden. Wer nicht an dieser Art Kriegsführung teilnehmen will, kann im größtenteils sicheren Raum der NPC-Fraktionen fliegen. Das erlaubt unterschiedlichsten Spielertypen, am selben Universum teilzunehmen und erzeugt eine bedeutungsvolle soziale Umgebung, wo Menschen mit völlig unterschiedlichem Hintergrund und Spielstil miteinander interagieren.

Der ständig drohende Verlust des **hart erarbeiteten Raumschiffes** verleiht sozialen Interaktionen zwischen Spielern in Eve Online eine enorme hohe Bedeutung.



Die Marktübersicht in Eve bietet **ausgefeilte Analyse-Tools** für Angebot und Nachfrage in der komplett von den Spielern getriebenen Wirtschaft.

Marktkräfte

Als die Wirtschaft erblühte und immer mehr Spieler hinzukamen, wurde es zunehmend schwieriger, die wirkenden Kräfte zu verstehen. CCP entschied sich, einen Wirtschaftswissenschaftler zu Rate zu ziehen. Der sollte im realen Leben erprobte ökonomische Prinzipien auf die Wirtschaft im Spiel anwenden und darüber berichten, so dass den Spielern mehr Daten zur Verfügung stehen und sie besser für die Zukunft vorausdenken können. Die Spieler brauchen diese Informationen, um kommende Aktionen ihrer Corporations zu planen. In Eve ist es nicht ungewöhnlich, dass Unternehmungen sechs bis zwölf Monate im Voraus geplant werden. Zudem sind Infos über Ökonomie, Business und gegenwärtige Trends einfach spaßig zu lesen. Diese Abteilung von CCP ist beständig gewachsen, und wir haben ein spezielles Team von Experten, die Daten zusammentragen sowie verschiedene Aspekte des Marktes und der politischen Landschaft in Eve beobachten.

Erst kürzlich gab es wieder einen macro-emergent moment zwischen CCP und den Spielern. Als nämlich die Bevölkerung auf eine gewisse Größe stieg, wurde es immer schwieriger, mit unseren Kunden zu kommunizieren. Zwar analysierten wir weiterhin die Meinung der Spieler mit Hilfe traditioneller Methoden, wie Forum, Kunden-Feedback, Messebesuchen, Marktforschung und den bier-unterstützten Gesprächen auf unserem Fanfest jeden November. Aber dabei verschafften sich vor allem die lautstärksten Kritiker und die glühendsten Fans von Eve Online und CCP Gehör. Dadurch war es manchmal schwierig, festzustellen, welche der individuellen Meinungen am meisten Zustimmung bei allen Spielern findet. Es war klar, dass wir die einzigartige Möglichkeit hatten, eine neue Art der Interaktion mit unseren Kunden zu etablieren, um deren Erfahrungen, Sorgen und Ideen für die Zukunft unserer virtuellen Welt besser zu verstehen. Das »Council of Interstellar Management« übernimmt nun diese Rolle (Genaueres dazu später im Artikel).

Von der Mikro zur Makro-Ebene

Auf der Mikro-Ebene kann man ebenfalls jeden Tag emergent behaviour beobachten. Wenn ein Spieler oder eine Corporation einen neuen Spielstil erforscht oder versucht, eine lange etablierte Spielmechanik zu durchbrechen, dann zeigen sich die Vorzüge von Eves Sandbox-Design. Häufig schaffen diese Mikro-Innovationen den Sprung auf die Makro-Ebene, wenn sich wirtschaftlicher oder militärischer Erfolg einstellt. Das hat sich etwa in den abgründigsten und abwegigsten Formen der Spionage zwischen Corporations manifestiert, außerdem in neuen Arten der Flottenkriegsführung, dem Fein-Tuning der Schiffsausrüstung oder jahrelangen Forumsschlachten um die Gunst der Leser – um nur wenige Beispiele zu nennen. Als Entwickler waren wir anfangs wirklich überrascht, wie viel Erfindungsgeist unsere Spieler, trotz der minimalen Werkzeuge, die wir Ihnen mitgaben, zeigten. Mit der Zeit wich die Überraschung einer zunehmend symbiotischen Beziehung, bei der CCP den Innovationen der Spieler, eine Struktur bietet, um sie ins Spiel zu übernehmen und in der uns die Spieler fortwährend dazu bringen, das Spiel in eine wahrhaftige virtuelle Welt zu verwandeln.

Unterm Strich haben wir herausgefunden, dass die Spieler die gleiche Leidenschaft für das Konzept von Eve empfinden, wie wir es anfangs als Erschaffer von New Eden hatten. Sie wollen

annähernd völlige Freiheit im Gameplay. Sie schließen sich ganz natürlich zu größeren und komplexen sozialen Strukturen zusammen – für mehr Sicherheit, Profit und Freundschaft. Unsere Spieler sind Teil der organischen, wachsenden Eve-Geschichte, wie man es sonst in keinem anderen Titel findet. Sie schwelgen in dem Gefühl der Selbstbestimmung, das aufgrund der Möglichkeit entsteht, Eve auf bedeutsame Art und Weise mit zu beeinflussen. Sie arbeiten, um beständige Änderungen der Spielwelt hervorzurufen. Viele streben danach, in einer Gemeinschaft von einer Viertel Million Menschen berühmt zu werden. Und sie können es. Eve bietet die Möglichkeit für Helden genauso wie für Bösewichter in einer fast moralfreien Umgebung. Viele dieser hochmotivierten Spieler sind schon genauso ein Teil der Geschichte von Eve wie die fiktionalen Charaktere, die unsere professionellen Autoren ersonnen haben.

Council of Interstellar Management

Da wir den guten Draht zu unserer stetig wachsenden Spielergemeinschaft nicht verlieren wollten, wurde das Council of Interstellar Management eingeführt. Die Idee dahinter ist es, ein demokratisch gewähltes Team aus Repräsentanten zu haben, das die Aufgabe hat, Anliegen der Spieler zu sammeln und diese dem Entwicklerteam zu präsentieren. Die ersten Wahlen fanden im Mai 2008 statt und das erste reale Treffen organisierten wir im Juni in unserem Heimatbüro in Reykjavik.

Die den Wahlen vorausgegangenen Kampagnen der Bewerber erinnerten dabei wenig überraschend an die unterschiedlichen Spielstile der Kandidaten. Manche setzten erfolgreich ihre Berühmt- oder Berüchtigtkeit ein, andere fingen relativ unbekannt an, fanden aber Anklang, indem sie bestimmte Spielergruppen wie die Industrialisten, die PVP-Spieler oder die Allianzen repräsentierten. Die neun Spieler mit den meisten Stimmen bekamen einen Sitz im Council of Interstellar Management (CSM) – derjenige mit den meisten Stimmen wurde Vorsitzender.

Zum ersten Treffen ließen wir die Mitglieder des CSM nach Island einfliegen, damit sie persönlich mit den Entwicklern reden konnten. Die ganze Angelegenheit wurde von unserem unabhängigen Moderator Agust H. Ingthorsson begleitet, der Direktor des Büros für Forschungsbeziehungen an der Universität von Island ist und Erfahrung mit Verhandlungen bei der Europäischen Union hat. CCP erachtete es als nötig, eine unabhängige Person bei den Treffen mit dabei zu haben und es half spürbar, sich auf die wirklichen Anliegen zu konzentrieren.

Unterm Strich waren die Meetings mit dem CSM ein Erfolg. Im Verlauf der drei angesetzten Tage sorgte ein solider Zeitplan dafür, dass die Treffen die gewünschten Ergebnisse brachten. Die einzige Unterbrechung war ein Ausflug an die Geburtsstätte der isländischen Demokratie, einen Ort namens Thingvellur. CCPs Ansatz für das Treffen war es, dem Council soviel Freiraum wie möglich zu geben, um ihre Anliegen als Repräsentanten der Spielerschaft zu formulieren und darzulegen. Genauso wichtig für die Unabhängigkeit und die Bestätigung des Einflusses des CSM ist dessen Verantwortung dafür, die Ergebnisse der Treffen an die Spieler zu berichten und aktuelle Angelegenheiten zur Agenda des nächsten Treffens hinzuzufügen.

Das »Council of Interstellar Management« ist eine **demokratisch gewählte Spielervertretung**.

Das CSM ist das Produkt von CCPs symbiotischer Nähe zu der Spielergemeinschaft von Eve und gleichzeitig dient es einem unserer übergeordneten Ziele – die Zukunft des Gaming zu erforschen. Welche traditionellen Barrieren der Interaktion mit den Spielern sollen wir in einer virtuellen Gesellschaft mit 250.000 Mitgliedern durchbrechen? Welche Schritte sollten wir machen, um weiterhin ein Spiel zu bleiben, ohne den kreativsten Teil von Eve Online zu behindern, nämlich unsere Spieler? Da die Spieler untrennbar mit der Eve-Erfahrung verbunden sind, widmen wir ihnen genauso viel Gedanken wie der PVP-Balance oder dem Schiffsdesign.

Seit dem Beginn von Eve war klar, dass es die Community ist, die ein Single-Shard-Universum antreibt. Der Spieler erschafft die Geschichte der persistenten Welt – auch wenn es nur ein Schürfer ist, der Mineralien für den Markt zur Verfügung stellt. Ohne die aktive Teilnahme der Spieler würde ein Großteil von Eve gar nicht existieren – dank der Spieler jedoch wird es eine dynamische, lebendige Ökonomie.

Alltägliches

Nicht alle von CCPs Interaktionen mit der Community basieren auf einem theoretischen Ideal oder hängen vom Single-Shard-Design ab. Genauso wie jedes andere Online-Rollenspiel haben wir Teams, die sich um Kundenservice und Community Management kümmern. Es ist sehr wichtig für unser geschäftliches Überleben, dass die Spieler merken, dass wir unser Pay-for-Service-Modell sehr ernst nehmen. Daher ist es unsere Aufgabe, ihr Vertrauen zu gewinnen und das Spiel fortwährend zu verbessern. Dafür hat sich ein System aus zwei kostenlosen Erweiterungen pro Jahr als sehr geeignet herausgestellt, denn uns ist klar, dass wir Eve selbst für unsere loyalsten Fans stets neu halten müssen.

Auf dem jährlich stattfindenden **Fanfest** pflegen die Entwickler besonders engen Kontakt zu den Spielern.

Das Community Team ist verantwortlich für den Großteil der Interaktion zwischen den Spielern und CCP. Es moderiert das Forum, betreut die Fanseiten und bringt Neuigkeiten über die Webseite und den Newsletter zu den Spielern. Allerdings funktioniert ihre Arbeit nicht als Einbahnstraße von oben nach unten, sondern sie nehmen ihre Aufgabe als Repräsentanten der Spieler sehr ernst. Sie sind eine unschätzbar wertvolle Informationsquelle für den Rest unserer Firma und werden bei Design- oder Politikänderungen stets zu Rate gezogen.

Der andere große Teil von CCP, der möglichst nahe an den Spielern bleibt, ist das Kundenservice-Team. Dazu gehören vor allem die Game Master in unseren drei Büros in Reykjavik, Atlanta und Shanghai. Die arbeiten in Schichten an den aufgelaufenen Petitionen der Spieler und greifen dabei auf umfangreiche Server-Logs zurück, um schnelle und effektive Lösungen für Probleme zu erreichen. Auch für die Spieler, deren Muttersprache nicht Englisch ist, haben wir extra Game Master angeheuert, die diese Sprachen beherrschen. Wie das Community Team wird auch der Kundenservice häufig von anderen Unternehmensteilen zu Rat gezogen und gehört meist zu den ersten, die das Operations-, Programmier- oder Design-Team bei Problemen alarmieren.

Schließlich gibt es noch die Abteilung für innere Angelegenheiten, die einzigartige Tools zur Verfügung hat, um sicherzustellen, dass die CCP-Angestellten sich korrekt verhalten. Das beinhaltet etwa Spielerbeschwerden über Game Master, aber auch das Überwachen von Mitarbeitern, die privat Eve Online spielen. Das stellt sicher, dass sie sich keine unfairen Vorteile verschaffen. Sollte ein Mitarbeiter die Unantastbarkeit von Eve in Gefahr bringen oder den Spielern den Eindruck geben, seinen Status als Entwickler zu missbrauchen, wird der Verantwortliche überwacht und gegebenenfalls zur Rechenschaft gezogen. Das ist sozusagen unsere Version der Börsenaufsicht inklusive einer FBI-Einsatzgruppe. Da das Wohlergehen all unserer Mitarbeiter vom Erfolg von Eve abhängt, nehmen wir diese Verantwortung sehr ernst.

Das Fanfest

Wir freuen uns jedes Jahr auf den November, denn dann findet das Fanfest nahe unseres Heimatbüros in Reykjavik statt. Es ist in vieler Hinsicht der beste Ausdruck für unsere Wertschätzung der Community. Es ist ein Fest für das Spiel, aber mehr noch für die Fans, daher der Name. Von den PVP-Turnieren bis zu den Gesprächen am runden Tisch, von den Vorführungen bis hin zu den Lesungen: Der ganze Event dreht sich darum, die Beziehung zwischen unseren Fans, Eve Online und CCP zu feiern. Wir haben vor langer Zeit entschieden, einen sehr offenen Zugang zu unserem Entwicklerteam zu gewähren, um Bier und Geschichten im zentralen Bereich des Fanfestes zu teilen. Wir haben uns vorgenommen, möglichst offen mit unserer Community umzugehen und Spielern einen ersten Blick auf bekannte Probleme sowie zukünftige Vorhaben zu geben.

Für viele meiner Kollegen ist es das eine Mal im Jahr, wo sie die Eve-Realität voll spüren können und die Auswirkungen der Projekte erfahren, an denen Sie das gesamte Jahr gearbeitet haben. Es ist sicherlich eine Katharsis für ganz CCP und wir sind zwar jedes Mal etwas verkatert danach aber auch voll frischem Tatendrang für das neue Jahr.

Eine der emotionalsten Erfahrungen für neue CCP-Angestellte auf dem Fanfest ist es, zu spüren, wie riesig die Bandbreite des Spiels und seiner Community ist. Es ist erstaunlich, wenn man realisiert, wie viel die Fans investieren, um das Spiel, dass sie lieben, zu feiern. Nur wenige von ihnen leben in Island, so dass die meisten einen Flug und eine Unterkunft benötigen und zudem Urlaub nehmen müssen. Und das alles für ein Spiel. Für unser Spiel. Sie bringen all das auf, um Zeit mit uns zu verbringen. Und wir fühlen uns verpflichtet, dass es den Aufwand Wert ist. Zum Glück zahlt sich jeder Moment auf dem Fanfest für uns zehnfach aus.

Bei CCP sind wir uns vollkommen bewusst, dass jeder Spieler dieses Opfer auch bringt, wenn er sich einloggt, um Zeit mit uns zu verbringen. Wir versuchen, ihm diesen Aufenthalt so angenehm wie möglich zu machen. Wir gestalten unser geliebtes Spiel immer mit den Fans im Hinterkopf. Denn es sind diese Fans, die uns einen unschätzbaren Teil der Struktur und Geschichte von Eve geliefert haben. Wir haben bereits Pläne für die kommenden Jahre – immer mit den Fans im Sinn. Da in Eve jeder an einer einzigen persistenten Welt teilnimmt, ist das Spiel zu etwas geworden, das eine wahrhaft symbiotische und bedeutungsvolle Beziehung zwischen den Spielern selbst und zwischen den Spielern und CCP geschaffen hat. Dieses Konzept verdeutlicht am besten, wie wir mit unserer Community umgehen.

Dr. Eyjólfur Guðmundsson

WANNA FLY
FIRST CLASS?

keen

UPGRADE NOW!

Develop AAA console games for major publishers

www.keengames.com

Aufbau und Pflege einer Kundenplattform
Die Crytek-Community

Eddy Dew von Crytek erklärt, mit welchen Methoden der Frankfurter Spiele-Entwickler seine Community organisiert, die Fans an alltäglichen Arbeitsabläufen teilhaben lässt und sie langfristig bindet.

Eddy Dew

ist Lead Community Manager bei Crytek.

Eddy leitet seit diesem Sommer die Community-Management-Abteilung beim Frankfurter Entwickler Crytek. Sein Team und er betreuen hauptsächlich die Community-Webseiten für Far Cry und die drei Crysis-Titel. Der 22-jährige war zuvor sowohl als Senior Spezialist im Kundenservice von Blizzard Entertainment Europe als auch beim kostenpflichtigen Online-Wettkampf-Portal Tournament Gaming tätig. Bei dem Gemeinschaftsprojekt mit dem Half-Life-Studio Valve war Eddy unter anderem für die Entwicklung von Anti-Cheat-Systemen verantwortlich.

Im Internet dreht sich heute alles um Communities. Wer im Netz Erfolg haben will, muss um die Aufmerksamkeit der Community kämpfen, sie regelmäßig mit Informationen beliefern und auf ihre Bedürfnisse eingehen. Um die professionelle Organisation der Netz-Gemeinschaften kümmert sich das Community Management. Bei Computerspielen geht es dabei in erster Linie um die Kommunikation zwischen Fans und Herstellern und hier besonders dem Entwicklerteam. Aus meiner Erfahrung im Kundenservice einiger der weltweit größten Spielehersteller kann ich sagen: Die Beziehung zum Fan kann in ihrer Bedeutung gar nicht überschätzt werden.

Deshalb muss das Community Management heute auch bei Spieleentwicklern gleichberechtigt neben den anderen Abteilungen stehen, selbst wenn die Positionierung innerhalb der Firma oft schwer fällt. Community Management ist eben von allem getrennt und gleichzeitig auch in allem involviert.

Das Community-Einmaleins

Die Bildung einer gut aufgestellten Community-Management-Abteilung stellt neu gegründete Firmen vor schwierige Herausforderungen. Der Schlüssel zum Erfolg liegt in einem starken Team. Mit guten Leuten lassen sich selbst heikle Probleme leicht lösen. Die endgültige Größe der Abteilung hängt natürlich vom Spiel, vom Entwickler und vom Publisher ab, doch dabei sollte letztendlich nie vergessen werden, dass Community Management ein Rund-um-die-Uhr-Job ist, für den man nicht nur leidenschaftliche, sondern auch genug Leute braucht.

Natürlich muss auch die Kommunikation mit der PR und dem Marketing der eigenen Firma, genauso wie der des Publishers unbedingt funktionieren. Ein steter Informationsfluss in beide Richtungen ist essenziell. So wurde das Multiplayer-Spiels Crysis Wars beispielsweise von vielen Vorschlägen aus der Fan-Gemeinschaft beeinflusst. Auf Wunsch der Community haben wir unter anderem Features eingebaut, die im Orginal-Crysis noch fehlten, wie einen Auto-Downloader und ein automatisches Patch-System. Das waren sogar einige der ersten Probleme, der wir uns in der Produktion von Crysis Wars angenommen haben.

Come together now

Eine interessante und funktionierende Webseite ist die Grundlage für erfolgreiches Community Management, ob für ein bestehendes Unternehmen oder eine neu gegründete Firma. Wir von Crytek betreiben momentan zwei Community-Seiten. Zum einen Crymod.com, die seit vier Jahren online ist und immer noch stetig wächst. Hier geht es hauptsächlich um unsere Spiele im Allgemeinen sowie um Mods für die CryEngine und die CryEngine2. Die Community hat hier ein paar fantastische Projekte veröffentlicht. Einige Sachen haben uns sogar so gut gefallen, dass wir über die Mods neue Mitarbeiter für Crytek rekrutieren konnten.

Unsere zweite und jüngste Webseite heißt MyCrysis.com und ist erst rund drei Monate alt. Sie ist das offizielle Community-Portal für alle Crysis-Titel. Die Seite ist unserer Meinung nach übersichtlich und schnell zu navigieren, außerdem lässt sie sich von den Usern leicht individualisieren. Viele Community-Treffpunkte sind zu kahl, bieten fast keine Neuigkeiten oder sind schlicht langweilig. Und dann gibt es aufwändige Seiten, die nur so vor grafischen Details und Wissenswertem strotzen, aber auch völlig überladen sind. Das Ideal liegt irgendwo dazwischen. Die Seite sollte zugänglich bleiben und Informationen gezielt vermitteln, gleichzeitig muss die Community sehen, dass man auf ihren Input reagiert. Über ein offizielles Community-Portal kann man den Fans zudem etwas ganz Besonderes bieten, nämlich eine Hintertür zu den Entwicklern: Fragestunden mit dem Team, Interviews mit den Produzenten und Live-Chats sind großartige, meist exklusive Möglichkeiten, um die Leute auf ein Produkt heiß zu machen.

Onliner offline treffen

Über MyCrysis.com betreuen wir unter anderem auch die offiziellen Foren – vielleicht der wichtigste Teil unserer Kommunikationsarbeit. In den Foren reden die Fans miteinander und vor allem mit uns. So schaffen wir die Grundlage für eine erfolgreiche Kundenbindung und damit für den Wachstum der Community. Unser Slogan für MyCrysis.com lautet: »Sammeln. Austauschen. Spielen.« Die Leute sollen sich über das Portal informieren, mit dem Crytek-Team unterhalten und mit anderen Fans über alle Aspekte der Crysis-Spiele diskutieren.

Neben dem direkten Foren-Support versuchen wir auch auf anderen Wegen, die Community zu unterstützen. Events und Messen gehören dabei zu den spaßigsten und gleichzeitig lohnenswertesten Möglichkeiten. Hier kann man richtig auftrumpfen. Unsere Teilnahme an der Games Convention 2008 in Leipzig war beispielsweise ein voller Erfolg. Die Fans hatten Spaß an unserer Show und wir konnten Demoversionen von zwei Spielen zeigen. Außerdem bieten solche Ereignisse auch immer die Gelegenheit, Community-Mitglieder im persönlichen Gespräch kennen zu lernen.

Hypes Marke Eigenbau

Aber was nützen Foren und Events, wenn man die Webseite nicht kontinuierlich mit spannenden Inhalten beliefert? News-Häppchen sind eine der großen Herausforderungen für Community Manager. Wir müssen uns nicht nur die Infos heranholen, wir sollten auch wissen, wie und in welcher Dosierung wir die Nachrichten an die Fans weiter reichen. Crysis war beispielsweise das Aushängeschild für die Cryengine2, also haben wir alles, was für das Spiel erstellt wurde, mit der Community geteilt. Das führte letztendlich aber dazu, dass einige Käufer das Gefühl hatten, bereits alles zu kennen. Bei Crysis Wars haben wir uns deshalb für eine andere Strategie entschieden: Wir gaben erst recht kurz vor dem Veröffentlichungstermin Informationen zum Spiel heraus, um so vor dem Verkaufsstart einen Hype zu generieren, der eine möglichst breite Masse der Fans und Medien erreicht. Praktischer Nebeneffekt: Wir konnten sehr viele finale Assets anbieten, vom Teaser-Clip über Screenshots und Wallpaper bis hin zum Intro-Video. Die Leute wurden so nicht mit Work-in-progress-Material überschüttet, dafür aber im Spiel mit neuen Inhalten belohnt.

Ressource: Mitmach-Gesellschaft

Viele Fans wollen sich in irgendeiner Form an Community-Portalen beteiligen. Dieses Interesse sollte das Community-Managment nutzten, denn Fans zu unterstützen ist generell immer eine gute Idee. So bieten die meisten Firmen ihre Foren zwar nur in englischer Sprache an, stellt man allerdings entsprechende Fansite Kits mit Artworks, Bannern und Templates bereit, werden sich engagierte Kunden meist selbst um die Lokalisierung kümmern und dabei gleichzeitig neue, internationale Communities erschließen. Wahrscheinlich werden sie sogar alle Infos und Neuigkeiten übersetzen, die man ihnen zuschickt. Eine Hilfe, auf die man als Community Manager nicht verzichten sollte.

Dabei darf man allerdings nicht die ausgewogene Balance zwischen Nehmen und Geben verlieren. Online-Wettbewerbe mit interessanten Preisen können hier den nötigen Ausgleich bieten. Einfache Anforderungen sind dabei meist die beste Wahl: Gestaltet ein Webseiten-Banner, entwerft eine Signatur oder einige Spielinhalte. So können alle Interessierten mitmachen, selbst wenn sie keines der Spiele besitzen, um die sich die Webseite dreht. Jeder sollte die Chance bekommen, tolle Preise zu gewinnen. Engagierte Forenmitglieder werden Wettbewerbe auch als Belohnung für ihre Loyalität wahrnehmen. Für diese Loyalität muss wiederum der Community Manager dankbar sein, denn diese Leute sind die Grundfesten der Fangemeinschaft – und ohne Fans geht gar nichts

Eddy Dew

Über seine neue Community-Plattform **MyCrysis.com** organisiert Crytek die offiziellen Foren, veranstaltet Online-Wettkämpfe und lädt zu Gesprächsrunden mit den Entwicklern ein.

Crymod.com ist seit vier Jahren die erste Anlaufstelle für **Mods zu Far Cry und den Crysis-Spielen.**

Von 2D zu 3D: Workshop Teil 4
Texturerstellung für ein 3D-Modell

Im vierten Teil unseres Workshops erstellen wir die Textur für unser modelliertes Haus aus den vorangegangenen Artikeln.

Chris Noeth

ist Art Director bei Rough Sea Games.

Der Diplominformatiker Chris Noeth arbeitet nach langjähriger Tätigkeit im Bereich Entertainment-Illustration seit 1999 als 2D-, 3D- und Pixel-Artist in der Spielebranche und unterrichtet an der Fachhochschule Würzburg-Schweinfurt zum Thema »3D-Modellierung und Animation« Informatik- und Designstudenten. Zu seinen Projekten zählen unter anderem die PC-Vollpreistitel Far West, Railroad Pioneer, Lula 3D und über dreißig Mobile Games wie Townsmen 2, Townsmen 3, Pirates Ahoy, Flitzer, Glory of the Roman Empire, Funky Monkey und My Model Train. Darüber hinaus ist er Comiczeichner der US-Comicserien »Tales from the Crypt« und »Ray Harryhausen Presents«.

http://blog.rough-sea.com

http://www.chrisnoeth.com

Nachdem wir in der letzten Ausgabe das Mapping für unser 3D-Modell angelegt haben, wenden wir uns im aktuellen Workshop der Erstellung der finalen Textur zu. Mit dem Mapping haben wir festgelegt, wie die Textur auf unser 3D-Modell projiziert wird. Daher können wir uns beim Erstellen der Textur auf die 2D-Ausarbeitung konzentrieren, ohne uns Gedanken um technische Details machen zu müssen. Als Ausgangsbasis dienen uns die beiden exportierten Bitmap-Grafiken mit den Mapping-Informationen aus dem vorhergehenden Workshop (Making Games 03/08, S. 30).

Vorbemerkung

Unser heutiger Workshop beschäftigt sich mit den Arbeitsschritten eines »Texture-Artists«, der dem Spielemodell eine ansprechende Oberfläche und Details verleiht. Das Erstellen von Texturen für Spielemodelle unterliegt hierbei einigen Einschränkungen im Vergleich zu Texturen für hochauflösende 3D-Modelle, wie sie zum Beispiel in gerenderten Filmen Verwendung finden. Spieletexturen besitzen durch den direkten Einsatz innerhalb eines Spiels die gleichen Performance-Einschränkungen wie 3D-Modelle, die sehr sparsam mit Polygonen modelliert werden müssen. Texturen müssen somit sehr klein und die zu mappenden Bereiche innerhalb einer Textur möglichst platzsparend angeordnet sein. Das gewährleistet eine gute Performance des Spiels auf möglichst vielen Computersystemen und nicht nur auf High-End-Rechnern. Die neuen Grafikkarten-Generationen erlauben allerdings die Verwendung immer größerer Texturen, weshalb wir unsere Grafiken mit einer Auflösung von 4096x4096 Pixeln beibehalten. Tipp: Es empfiehlt sich, Texturen immer in einer größeren Auflösung als der Zielgröße anzulegen und sie nach Fertigstellung herunter zu rechnen, da so wesentlich mehr eingearbeitete Details in der Textur erhalten bleiben. Für unseren Workshop arbeiten wir allerdings direkt in der Zielgröße, da die Erstellung auf älteren Rechnern sonst nur sehr schwer nachvollziehbar ist. Grundsätzlich gibt es unterschiedliche Möglichkeiten, Tex-

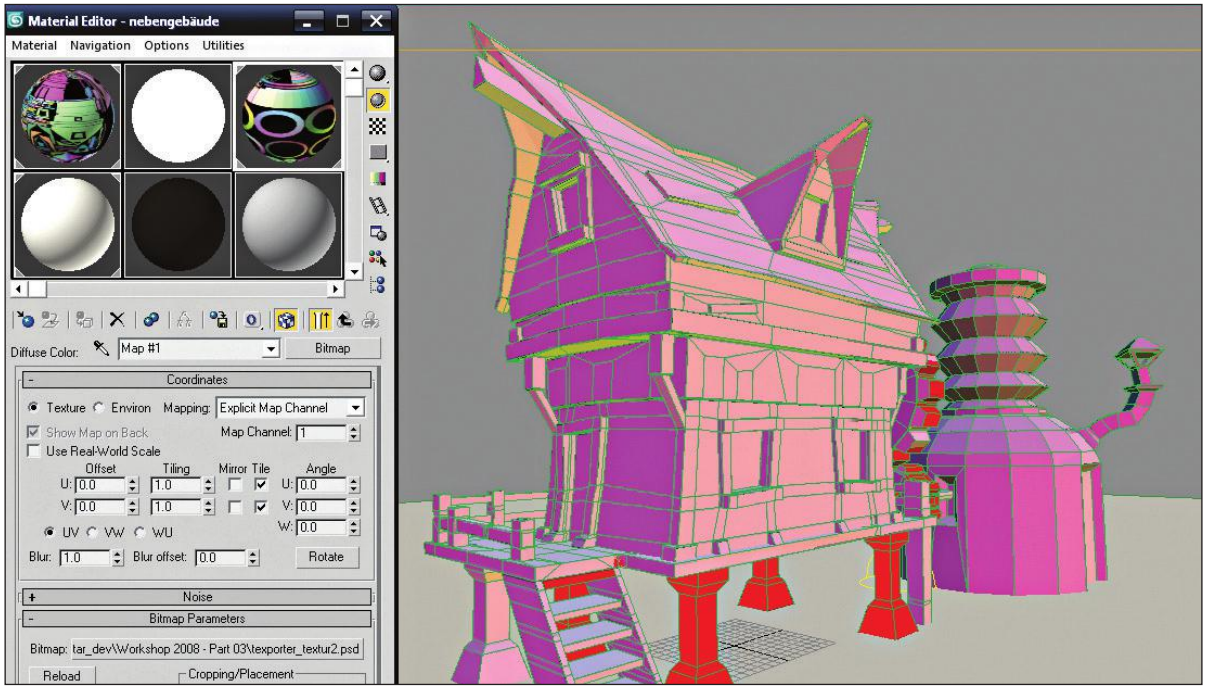


Abbildung 1: Ausgangsmodell mit »Texporter«-Bitmaps

turen für 3D-Modelle zu erstellen. Diese Möglichkeiten hängen von der Art und auch von der Verwendung des Modells ab. Handelt es sich etwa um ein hochauflösendes 3D-Modell für eine realistische Darstellung in einer Architekturszene, kann der Einsatz von Fototexturen sinnvoll sein, um möglichst originalgetreue Darstellungen zu erzeugen. Für Spielemodelle gelten jedoch andere Regeln, da diese aus Performancegründen meistens nur wenige Polygone besitzen, und die Details dann über die Textur gezielt in die Spielwelt integriert werden. Daher ist in diesem Fall der Einsatz von Fototexturen nicht oder nur begrenzt sinnvoll. Einige Spiele nutzen trotzdem eine Vielzahl an Fototexturen, allerdings wirkt die Welt dieser Titel in den meisten Fällen sehr überladen, nicht auf die enthaltenen Elemente abgestimmt und insgesamt wenig ansprechend.

Ein Texture-Artist muss sich bewusst sein, dass er niemals nur eine Textur für ein einzelnes Modell erstellt, sondern jederzeit die geplante Spielwelt und die Wirkung der Einzeltextur innerhalb derselben berücksichtigen muss. Daher sollte jede zum Einsatz kommende Textur im Einklang mit der Spielwelt stehen und gleichzeitig ansprechend gestaltet sein.

Der Unterschied zwischen Shader und Textur

Beim Erstellen unserer Textur sollten wir zwischen den Begriffen »Textur« und »Shader« unterscheiden, da die beiden oft verwechselt werden. Der »Shader« definiert die komplette Oberflächendarstellung eines 3D-Modells innerhalb eines Spiels. Dies schließt auch Effekte wie Spiegelungen, Bump-Mapping oder Lichtabsorption ein. In unserem Workshop steht der Begriff »Textur« für ein Bild, welches über die Oberfläche des Modells projiziert wird. Für alle zusätzlichen Eigenschaften der Oberflächenbeschaffenheit sind spezielle Texturen notwendig, welche die Shader-Eigenschaften definieren. Für die Spiegelungen und das Bump-Mapping werden zum Beispiel Graustufenbilder verwendet, die je nach Grauwert die Stärke der Spiegelung oder des Bump-Mappings für bestimmte Bereiche des Modells festlegen. Diese »Shader-Texturen« werde ich im Workshop nicht gesondert besprechen, da sie nach dem gleichen Prinzip erstellt werden.

Erstellen der Textur

Bevor wir mit dem eigentlichen Erstellen der Textur beginnen, müssen wir uns innerhalb eines Projektes zuerst ein Bild der Spielwelt machen. Dies geschieht meistens anhand von vorhandenen Concept-Artworks und den Anweisungen des Art Directors. Das ist wichtig, damit wir die Textur bestmöglich auf den Einsatz innerhalb der Spielewelt abstimmen können.

Schauen wir nun unser 3D-Modell an und stellen wir uns vor, es sei für den späteren Einsatz in einem typischen Fantasy-Online-Rollenspiel gedacht. Für einige Texture-Artists mag der erste Gedanke vor der Erstellung der Textur möglicherweise ihrem Fotoapparat gelten. Allerdings sollte man das aus den bereits genannten Gründen möglichst vermeiden. Unterstützend können Fotos bei der Texturerstellung hilfreich sein, aber Fotos sollten niemals eins zu eins in eine Textur eingebunden werden, denn das führt mit großer Wahrscheinlichkeit zu einem abstoßenden Ergebnis im Spiel. Texturen sollen in Spielen meist nicht absolut fotorealistisch sein, sondern eben ansprechend und »verspielt«. Aus genau diesem Grund müssen wir

Abbildung 2: Basisstruktur der Textur

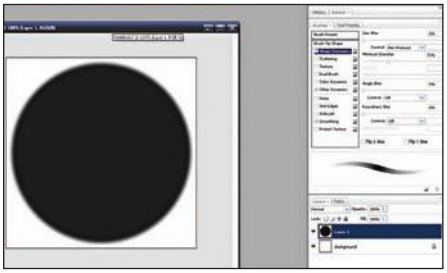


Abbildung 4: Eigene Pinselspitze in Photoshop

Abbildung 5: Drucksensitivität im Einsatz

uns zunächst Gedanken über die Welt innerhalb des Spiels machen. Im zweiten Schritt machen wir uns klar, was unser Modell in dieser Spielwelt repräsentiert, wo es platziert ist, wie die Umgebung aussehen wird und wie wichtig das Modell für diese Welt ist. Hiervon hängt dann beispielsweise der Detailgrad der finalen Textur ab. Einige gängige Vorüberlegungen können unter anderem sein:

Abbildung 3: Aktuelle Textur auf dem Modell

- Aus welchen Materialien besteht unser Spielobjekt?
- Was zeichnet diese Materialien aus?
- Handelt es sich um neue, alte, abgenutzte oder angegriffene Materialien?
- Ist die Umgebung eventuell feucht, gibt es Schimmelflecken, etc.?

Im Falle unseres 3D-Modells handelt es sich um ein Holzhaus, das in einer wüstenähnlichen Landschaft steht. Das Gebäude ist etwas älter und bereits etwas mitgenommen. Beginnen wir nun schrittweise mit der Ausarbeitung unserer Textur. Hierzu verwenden wir die am Ende des letzten Workshops exportierten Bilddateien.

Erstellen der Textur: Schritt 1

In Abbildung 1 sehen Sie das Ausgangsmodell mit den beiden zugewiesenen Bitmap-Dateien, welche wir im vorangegangenen Workshop erstellt haben. Die Hilfstextur für das Mapping aus dem letzten Workshop wurde im Materialeditor durch die beiden exportierten Bitmaps ersetzt. Über die in den Bitmaps angezeigten Geometrieteile erstellen wir im Folgenden die Finaltexturen. Zu Beginn malen wir grob eine Basisstruktur für die wichtigsten Bereiche, wie etwa die Wände. Bitte legen Sie hierfür einen neuen Layer in Photoshop an. Diese Struktur soll die spätere Bretteranordnung zeigen und dient gleichzeitig als Basis für die weitere Ausarbeitung. In Abbildung 2 sehen Sie einen Teilbereich der Basisstruktur. Sobald Sie diesen Zwischenstand abspeichern, wird die Textur automatisch in 3ds Max nachgeladen und auf dem Modell angezeigt. Ein gerendertes Zwischenbild des Modells sehen Sie in Abbildung 3.

Erstellen der Textur: Schritt 2

Für die weitere Ausarbeitung der Textur benötigen wir einen möglichst weichen Pinsel, damit

Abbildung 6: Pinseltest: Basisfarben / Brettertextur

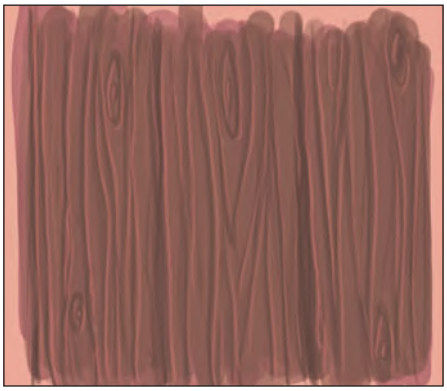


Abbildung 7: Lichteinfall / Brettertextur

Abbildung 8: Details / Brettertextur

wir durch den Stiftandruck und den so erzeugten transparenten Farbauftrag Details erstellen können. In Abbildung 4 sehen Sie eine in Photoshop neu erstellte runde Pinselspitze mit weichem Rand. Um mit dieser Pinselspitze einen weichen Farbauftrag zu erzeugen, stellen Sie deren Wert für »Flow« auf circa 20 Prozent. Wenn Sie mit einem drucksensitiven Zeichentablett wie beispielsweise dem Wacom Intuos 3 arbeiten, sollten Sie mit der neu angelegten Pinselspitze problemlos in der Lage sein, einen Farbauftrag wie in Abbildung 5 zu erstellen.

Erstellen der Textur: Schritt 3

Testen Sie nun die neue Pinselspitze beim Anlegen einer unabhängigen Bretterstruktur. Beginnen Sie mit einer dunklen Basisfarbe und den Bretterformen, wie in Abbildung 6 gezeigt. Fügen Sie dann beleuchtete Bereiche hinzu, um den Brettern etwas mehr Tiefe und Struktur zu verleihen. Durch den ungleichmäßigen Farbauftrag, welcher von der Andruckstärke des Stifts abhängt, erzeugen Sie zusätzliche Details bei Strichüberlagerungen, welche die Bretterstruktur verstärken. In Abbildung 7 sehen Sie diesen Zwischenschritt. In einer Überarbeitung der bisherigen Bretter fügen Sie am Ende noch mehr Details, wie zum Beispiel Kratzer im Holz, hinzu. Das Ergebnis sehen Sie in Abbildung 8.

Erstellen der Textur: Schritt 4

Mit den erprobten Techniken malen Sie nun über die erstellte Basisstruktur in der angewandten Textur weitere Details in die Holzbretter. Abbildung 9 zeigt ein entsprechendes Beispiel. Abbildung 10 zeigt einen Bildausschnitt unseres Gebäudes, bei dem die Basisstruktur der Wandbretter zu sehen ist. Hier wird auch deutlich, weshalb ein gutes Mapping unerlässlich ist – nur so lassen sich genaue Übergänge an den Kanten des Modells erzeugen.

Um gemalte Texturbereiche am Modell zu testen, sollten Sie in kurzen, regelmäßigen Abständen Testbilder rendern. Da das wiederholte

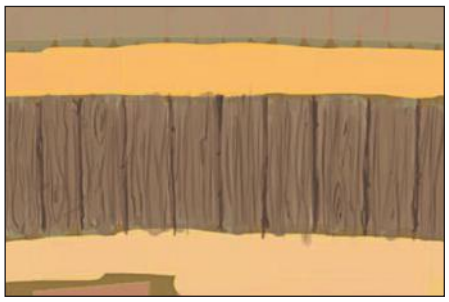


Abbildung 9: Ausarbeitung der Bretter

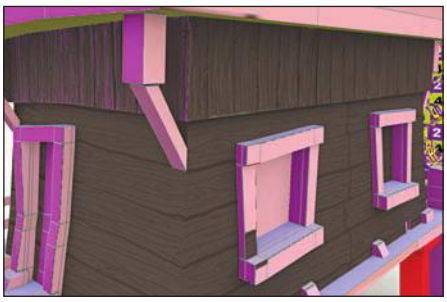


Abbildung 10: Aktuelle Brettertextur auf dem Modell

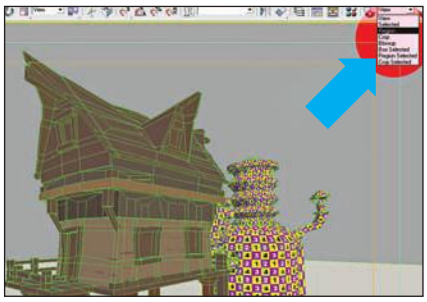


Abbildung 11: Rendern eines Bildausschnitts in 3ds Max

Abbildung 12: Gerenderter Bildausschnitt in 3ds Max

Abbildung 13: Texturdarstellung mit Texporter-Geometrie

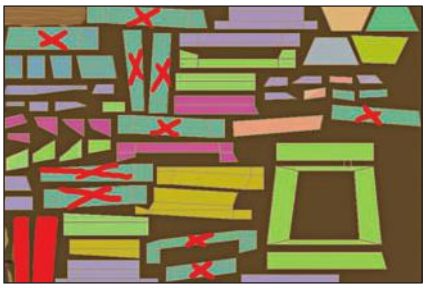


Abbildung 14: Texturmarkierungen

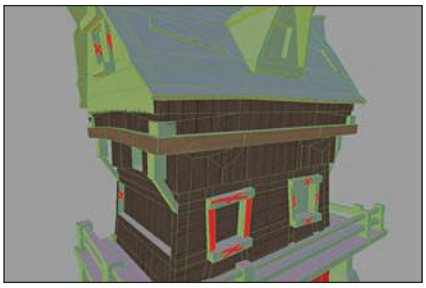


Abbildung 15: Texturplatzierung

Rendern des kompletten Modells in diesem Fall meist überflüssig ist und viel Zeit in Anspruch nimmt, verwenden Sie am besten die in 3ds Max angebotene Möglichkeit, einen Bildbereich (»Region«) zu rendern. Diesen Bildbereich können Sie sich durch Auswahl mit der Maus selbst festlegen. **Abbildung 11** zeigt die Auswahl von »Region« für das Rendering in 3ds Max. In **Abbildung 12** sehen Sie das Render-Ergebnis solch eines festgelegten Bildbereichs.

Erstellen der Textur: Schritt 5

Da durch das Übermalen und Hinzufügen von Details mittlerweile die Ränder der Objektgeometrien nicht mehr zu erkennen sind, sollten Sie sicherheitshalber immer unterschiedliche Layer anlegen, bei denen Sie jeweils nur Teile der Geometrien bearbeiten. Behalten Sie daher immer in einem Layer die ursprünglich exportierte Bitmap-Grafik mit den Geometrien bei. Die **Abbildung 13** zeigt dies im Beispiel.

Sollte es dennoch an Stellen etwas unübersichtlich werden und Sie wissen nicht, wohin etwa eine bestimmte Geometrie gehört, dann hilft ein einfacher Trick: Zeichnen Sie farblich auffällige Kreuzchen und Markierungen in die Geometrien, deren Zugehörigkeit für Sie nicht erkennbar sind **(Abbildung 14)**. Ein gerendertes Bild **(Abbildung 15)** vermittelt dann schnell die richtige Platzierung.

Finalisieren der Textur

Sie kennen nun die wichtigsten Schritte in der Ausarbeitung der Texturen. Setzen Sie die besprochenen Schritte wiederholt für die restlichen Teile des Modells ein. Berücksichtigen Sie die unterschiedlichen Materialeigenschaften des Modells und versuchen Sie die fortgeschrittene Textur farblich anzugleichen, damit sich einzelne Teile nicht zu sehr voneinander abheben. In **Abbildung 16** sehen Sie eine Zwischenstufe der Textur am Modell. **Abbildung 17** und **Abbildung 18** zeigen eine aufgehellte Textur mit zusätzlicher Struktur, um damit zu testen, ob diese Textur eventuell mehr Details vermittelt.

Abbildung 19 demonstriert den Zwischenstand mit der ausgewählten Textur, welche für die weitere Ausarbeitung Verwendung findet.

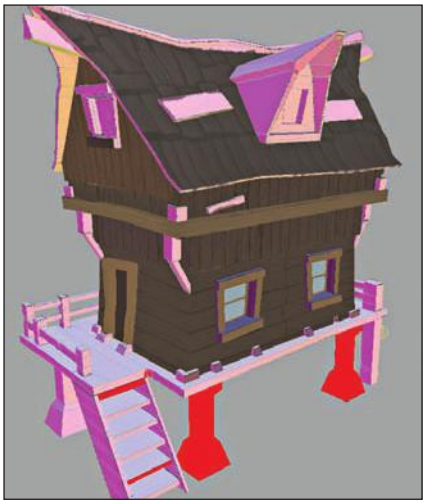


Abbildung 16: Aktuelle Textur auf dem Modell

Abbildung 17: Aufgehellte Testtextur

Abbildung 18: Testtextur auf dem Modell

Abbildung 19: Aktuelle Textur für weitere Ausarbeitung

Abbildung 20: Beginn der Detailausarbeitung (Dach)

Abbildung 21: Aktuelle Textur mit begonnenen Details

Abbildung 22: Dachdetails am Modell

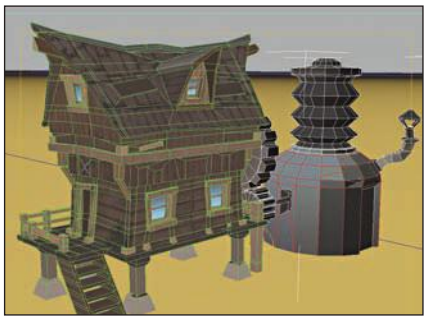


Abbildung 23: Haus mit Nebengebäude

An dieser Stelle kann mit der Einbindung von Details in die Textur begonnen werden. Abbildung 20 zeigt dies am Beispiel des Hausdachs. Durch eine detaillierte Ausarbeitung der Dachbretter mit Licht und Schatten, Rissen und Kanten erzeugen Sie allein über die Textur den Eindruck eines plastischen Dachs mit Höhen und Vertiefungen. In Abbildung 21 finden Sie eine Zwischenstufe der Texturausarbeitung und in Abbildung 22 sehen Sie das texturierte Dach. Im abschließenden Schritt bearbeiten Sie die zweite Textur für das Nebengebäude. Die runde Objektgeometrie des Modells erfordert zu Beginn eine Einstellung der richtigen Glättungsgruppen, um das Nebengebäude am Ende auch rund wirken zu lassen. Hierzu wählen Sie mittels des Modifikators »Edit Mesh« die entsprechenden Flächen im Modell aus und fügen die passende Glättungsgruppe hinzu (diese finden Sie am Ende des Rollout-Menüs von »Edit-Mesh«). Abbildung 23 zeigt das komplette Modell bestehend aus dem Haupthaus samt dem Nebengebäude. In Abbildung 24 sehen Sie unsere fast fertige Textur im Einsatz. Es fehlen lediglich noch ein paar letzte, ausschmückende Details.

Abbildung 24: Aktueller Zwischenstand

Fertige Texturen

Abbildung 25 und Abbildung 26 zeigen die finalen Texturen mit den hinzugefügten Details wie etwa der Grünpflanze oder den Schwertern über der Eingangstür des Hauptgebäudes. Solche Details lassen sich durch den richtigen Einsatz von gezeichnetem Licht und Schatten relativ einfach über die bisherige Textur malen und tragen sehr viel zum Erscheinungsbild des Modells bei. Im Spiel erkennt man nur aus nächster Nähe und aus einem bestimmten Winkel, dass es sich hier um gemalte Texturdetails handelt. Das Aufmacherbild dieses Workshops zeigt das 3D-Modell mit den finalen Texturen.

Chris Noeth

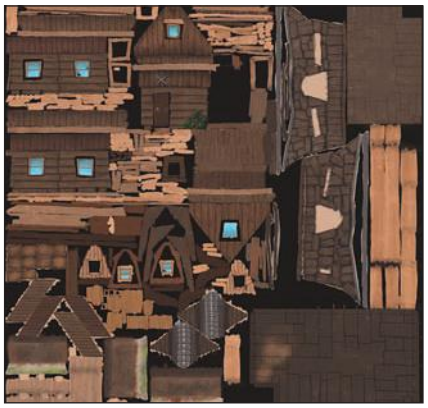


Abbildung 25: Finale Textur des Hauptgebäudes

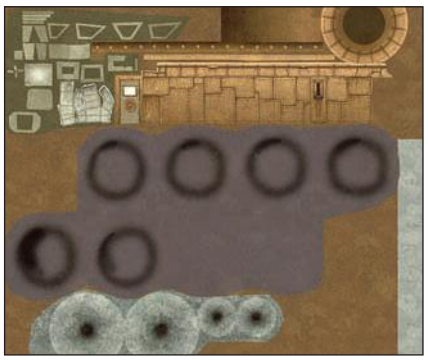


Abbildung 26: Finale Textur des Nebengebäudes

Bewegung in zwei Dimensionen
2D-Animationen in The Whispered World

Gunnar Bergmann, Franziska Reinhard und Fabian Pahl von Daedalic Entertainment erklären die Entstehung und Umsetzung von zweidimensionalen Grafik-Animationen ihres Fantasy-Adventures The Whispered World.

Abbildung 1: Der Held des Spiels durchlief mehrere **Roh-Phasen** (»Roughs«) bis zu seiner finalen Gestalt.

Gunnar Bergmann

ist Character-Animator bei Daedalic Entertainment.

Der Graphic Artist Gunnar Bergmann ist einer der Charakter-Animatoren bei The Whispered World. Der Diplom-Designer hat zuvor an Animationsfilmen wie Das doppelte Lottchen und Derrick sowie an TV-Serien und Comics mitgearbeitet.

Franziska Reinhard

ist Game-Designerin bei Daedalic Entertainment.

Franziska Reinhard hat während ihres Diplomstudiums an der Fachhochschule Lübeck ein komplettes 2D-Adventure entwickelt. Bei The Whispered World ist die Game Designerin jetzt zuständig für Rätseldesign und -scripting.

Fabian Pahl

ist Senior Programmer bei Daedalic Entertainment.

Wenn der Medieninformatiker Fabian Pahl nicht gerade im offensiven Mittelfeld beim MTV Laßrönne 1919 schaltet und waltet, entwickelt er als Senior Programmer bei Daedalic die Tools und die Engine für The Whispered World.

Ein Spiel, das komplett auf handgezeichnete 2D-Grafiken baut, stellt die Entwickler vor spezielle Herausforderungen. Das Fantasy-Adventure The Whispered World setzt auf Figur-Animationen in Trickfilm-Qualität und 2D-Hintergründe mit komplexem Parallax-Scrolling. Die enormen Texturmengen, die dabei im Grafikspeicher verwaltet werden müssen, bedeuten auch für die Programmierung besondere Probleme. Einige der interessantesten Schritte, um Protagonist Sadwick flüssig durch die detaillierten Fantasy-Hintergründe zu bewegen, erläutert dieser Artikel.

Character Animation

Das erste Stadium beim »Erwecken« eines 2D-Charakters, in unserem Fall Sadwick, dem Helden von The Whispered World, sind die Konzeptentwürfe der Figur. In groben Skizzen, den so genannten »Roughs«, werden zunächst der Look und die Eigenschaften sowie das Farbdesign der Figur festgelegt. Sadwick durchlief dabei zahlreiche Mutationen, vom kugeligen Clown bis zum jugendlichen Gaukler in erdigen Farbtönen (Abbildung 1). Steht das Character-Design, wird ein »Turnaround« des Charakters erstellt. Das bedeutet, der Charakter wird von allen fünf Ansichts-Positionen (vorne, diagonal vorne, seitlich, diagonal hinten und hinten) gezeichnet (Abbildung 2). Das gewährleistet eine gleich bleibende Proportionalität und einen einheitlichen Look. Der Turnaround dient den Animatoren als Basis für ihre Arbeit. Sind alle Parameter des Charakters festgelegt, beginnen die Animatoren mit den Bewegungen.

Dazu gehört allen voran ein so genannter »Walkcycle«, eine Schleife von Einzelbildern, die den Gang der Figur simuliert, ohne dass diese sich von der Stelle rührt (Abbildung 3). Wird nun der Bildhintergrund hinter der Figur horizontal verschoben, entsteht der Bewegungseindruck. Natürlich gibt es auch Animationen, in der die Figur sich vor einem feststehenden Hintergrund fortbewegt. In diesem Fall wird der Charakter tatsächlich innerhalb der dafür vorgesehenen Parameter von A nach B geschickt.

Bei The Whispered World entsteht jedes Einzelbild zunächst traditionell an einem Leuchttisch – in zeichnerischer Handarbeit, ohne vom Computer errechnete und eingefügte Zwischenschritte. Während beim Trickfilm normalerweise 24 Bilder pro Sekunde ablaufen, sind es

Abbildung 2: Im **Turnaround** wird der Charakter von allen fünf Ansichts-Positionen gezeichnet.

Abbildung 3: Der **Walkcycle** zeigt in einer Bilderschleife an, wie sich die Figur bewegen soll.

in einem Spiel für gewöhnlich zwölf Bilder pro Sekunde. Das kann je nach Komplexität der Animation variieren, aber allgemein gilt bei Spiele-Animationen: so viele Bilder wie nötig, so wenig wie möglich. Immerhin bedeutet jedes zusätzliche Bild entsprechend mehr Speicheraufwand.

Zu den Laufanimationen kommen bei The Whispered World zahlreiche weitere Standardanimationen wie Greifen, Nehmen oder Pausieren sowie Sondermanöver, die nur in bestimmten Kulissen vorkommen – Klettern, Springen und so weiter **(Abbildung 4)**.

Rough-Phase

An erster Stelle steht die Überlegung, wie sich die Figur in der zu animierenden Situation verhalten beziehungsweise bewegen soll. Der Animator übernimmt hier die Aufgabe eines Schauspielers, nur dass er nicht mit seinem Körper und seiner Mimik agiert, sondern diese auf dem Papier und in die Figur übertragen muss. Dazu werden zunächst schnelle Skizzen angefertigt. Diese Roughs sind design-unabhängig und legen allein den Bewegungsablauf fest. Mit diesen Skizzen kann der Animator die gesamte Animation planen und alle nötigen Parameter bestimmen: Dazu werden die Rough-Skizzen eingescannt und in Photoshop als Dokument mit einer Ebene pro Zeichnung angelegt.

Nun lässt sich eine erste Testanimation erstellen. In aktuellen Versionen bietet Photoshop dafür eine Animationsleiste, bei Versionen unter CS2 bietet ImageReady nahezu identische Möglichkeiten. Sind die einzelnen Frames der Animation in der Leiste platziert, wird das Timing, also die Anzeigedauer jedes Frames, bestimmt. Anhand dieser Testanimation kann nun korrigiert und verbessert werden, können Frames entfernt oder hinzugefügt werden, bis das gewünschte Ergebnis erzielt ist. Nachdem die fertige Rough-Animation und somit die Frame-Anzahl feststeht, wird noch das finale Timing festgelegt. Nachdem somit die Grundlage steht, kann die Ausarbeitung beginnen.

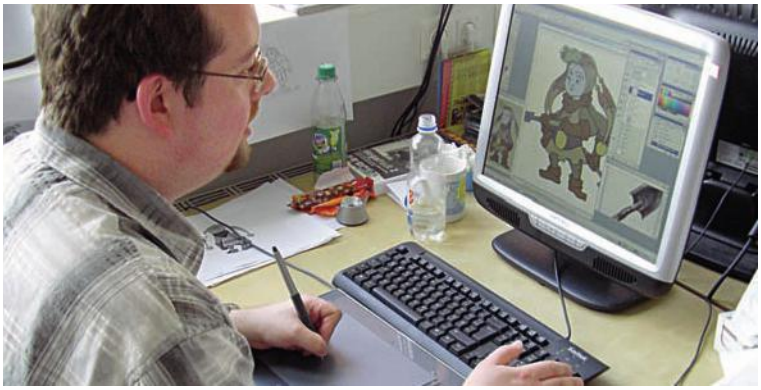


Abbildung 5: **Gunnar Bergmann** überarbeitet Sadwick am Grafiktablett.

Clean-Up-Phase

Bei kleineren Animationen bietet sich das Reinzeichnen direkt per Grafiktablett am Rechner an **(Abbildung 5)**. Die meisten Grafiker bevorzugen jedoch die klassische Variante: Die ausgedruckten Roughs werden auf den Leuchttisch gelegt und dann auf Animationspapier mit Stift oder Feder sauber nachgezeichnet **(Abbildung 6)**. Danach werden die Seiten eingescannt und als Ebenendokument in Photoshop angelegt. In beiden Fällen hat der Animator nun ein Dokument mit einer Ebene pro Clean-Up. Nachdem er Unsauberkeiten oder Verzeichner korrigiert hat, erstellt er eine neue Testanimation. Bei der finalen Linie können immer noch kleinere Unstimmigkeiten sichtbar werden, die bei den Roughs aufgrund der Grobheit der Zeichnungen nicht aufgefallen sind. Sind diese korrigiert worden, ist das Clean-Up abgeschlossen.

Kolorierungs-Phase

Die Kolorierungs-Phase lässt sich grob in drei Schritte aufteilen: Farbe, Schatten und (falls vorgesehen) Effekte **(Abbildung 7)**. Der Animator beginnt mit der Grundkolorierung der Figur. Um die Übersicht in der Ebenenpalette zu behalten, bietet es sich an, für jedes Einzelbild ein

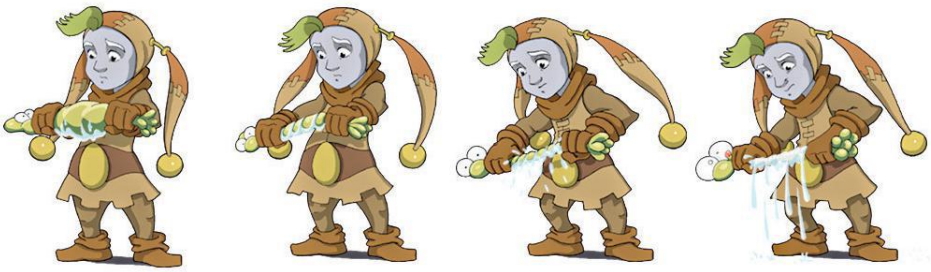


Abbildung 4: Neben Standardanimationen wie Aufheben oder Gehen beherrschen die Spielfiguren zahlreiche **Sondermanöver**.

Abbildung 6: Die Animatorin **Simone Kesterton** zeichnet Roughs ins Reine.

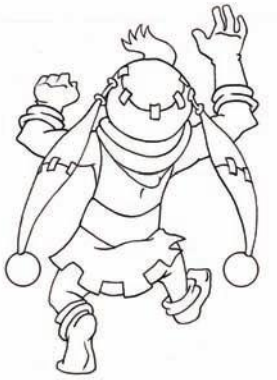


Abbildung 7: In der **Kolorierungs-Phase** werden Farben und Effekte zu den Umrissen hinzugefügt.

Ebenen-Set anzulegen, in das alle benötigten Ebenen samt der Clean-Up-Ebene gelegt werden. Dann wird zunächst die Clean-Up-Zeichnung freigestellt (»Outline«), sodass der Rest der Ebene transparent angezeigt wird. Darunter wird eine Farbebene gelegt und die Grundfläche zunächst mit der vorherrschenden Basisfarbe gefüllt. Dann werden die Flächen der Figur mit den entsprechenden Farben versehen. Beachten Sie dabei, dass die Flächen in der Outline-Ebene immer klar getrennt sind, da sonst eine schnelle und saubere Auswahl per Zauberstab-Werkzeug nicht möglich ist. Abschließend werden mögliche »Blitzer«, kleine freie Stellen, die von der Auswahl nicht erfasst wurden, manuell mit dem Pinsel-Werkzeug korrigiert.

Die Schatten werden auf einer gesonderten Ebene aufgetragen. Auch hier gibt es mehrere Möglichkeiten, wir bleiben aber bei der einfachen und effizientesten: Wir wählen die zu schattierende Fläche in der Outline-Ebene aus und bearbeiten sie in der Schatten-Ebene. Der Schatten lässt sich mit der gewünschten Schattenfarbe (in unserem Fall ein einfaches Schwarz, im Ebenenmenü auf 50% Transparenz gesetzt) bequem mit dem Pinselwerkzeug auftragen. Die Auswahl verhindert hierbei das Übermalen in andere Flächen. Den Standschatten unter der Figur legen wir einfach frei mit dem Pinsel an, ebenfalls in 50% Schwarz. Zum Schluss legen wir noch eine Effekt-Ebene an, auf die zum Beispiel Glanzpunkte für Augen oder reflektierende Flächen aufgetragen werden.

Zum Abschluss kontrollieren wir, ob alle Flächen sauber ausgefüllt sind oder ob die äußeren Ränder überschritten wurden. Durch eine Extra-Hintergrundebene in 100% Schwarz wird auch die kleinste Fehlstelle sichtbar. Wenn alles in Ordnung ist, fügen wir die Ebenen innerhalb des Ebenen-Sets zusammen. Diese Arbeitsschritte wiederholen wir für jedes Einzelbild beziehungsweise Ebenen-Set, sodass man am Ende wieder nur eine Ebene pro Bild hat. Jetzt muss noch der Arbeitsraum des Dokuments auf den kleinstmöglichen Nenner reduziert werden, um unnötigen Speicherplatz zu verhindern. Da jetzt alle Bilder mit transparentem Hintergrund übereinander liegen, ziehen wir einfach eine Auswahl um den sichtbaren Bereich und stellen diese frei. Die Bilder werden nun noch als PNG-Dateien abgespeichert – dieses Format gewährleistet die Transparenz bei der späteren Verarbeitung. Als Anhang erstellt der Animator noch ein Textdokument mit der Timingliste, der Darstellungszeit jedes einzelnen Frames in Sekunden. Mit dieser Information übergibt der Animator seine Arbeit an den Scripter, der die Animation ins Spiel einbaut.

Backgrounds

Die Hintergrundkulissen von The Whispered World werden ebenfalls komplett per Hand gezeichnet und mit zahlreichen Detail-Animationen versehen, die die Spielwelt zum Leben erwecken sollen. Alle Bildelemente, die sich während des Spielverlaufs manipulieren oder ins Inventar aufnehmen lassen, müssen dazu auf separate Ebenen gezeichnet werden. Das Gleiche gilt für Objekte, die animiert werden oder hinter denen die Spielfigur entlang gehen kann. Diese einzeln abgespeicherten Elemente werden in die Engine importiert und dort pixelgenau zu einer Kulisse zusammengesetzt. Jede Kulisse besteht mindestens aus einer Hauptebene, auf der sich der begehbare Bereich befindet, in dem sich Sadwick bewegen kann. Auf zusätzlichen Ebenen können Grafiken angeordnet werden, die vor oder hinter diesem Bereich liegen.

Scrolling

Überschreitet eine Kulisse die normale Bildschirmbreite, scrollt sie in The Whispered World nicht einfach weiter. Stattdessen benutzen wir eine Technik, die aus dem klassischen Animationsfilm stammt – das Parallax-Scrolling. Dabei entsteht durch das Verschieben mehrerer zweidimensionaler Hintergrundebenen in unterschiedlichen Geschwindigkeiten ein Eindruck von räumlicher Tiefe. Eine Kulisse von The Whispered World setzt sich aus bis zu 20 unabhängigen Ebenen zusammen **(Abbildung 8)**. Jede dieser Parallax-Ebenen besteht aus Einzelelementen, die großflächiger gezeichnet sind, als es zunächst den Anschein hat. Die Bildelemente überlagern sich gegenseitig, sodass nicht alles auf einen Blick zu sehen ist. Erst durch das Verschieben der Parallax-Ebenen werden verdeckte Elemente freigelegt.

Bei der Produktion einer Kulisse legen wir zunächst die Hauptebene an. Anschließend werden die Parallax-Ebenen erstellt, die in Z-Richtung vor und hinter ihr liegen. Für jede dieser Ebenen wird im Verhältnis zur Hauptebene eine individuelle Scroll-Geschwindigkeit bestimmt. Mit zunehmender Entfernung zur »Kamera« nimmt die Geschwindigkeit ab. Vordere Ebenen bewegen sich schneller als hintere und geben den Blick auf Bildelemente frei, die vorher nicht zu sehen waren. Beim Betrachter entsteht der Eindruck, als habe sich der Blickwinkel etwas verschoben. So kommt ein Gefühl für Tiefe auf, obwohl die Grafiken nur zweidimensional sind. Eine Kulisse wirkt durch diese fortlau-

Abbildung 8: Die Hintergründe in The Whispered World bestehen aus bis zu 20 **Parallax-Ebenen**, die die Illusion räumlicher Tiefe erzeugen.

fende Veränderung beim Scrollen weniger statisch. Bei The Whispered World setzen wir Parallax-Scrolling sowohl horizontal als auch vertikal und in jede andere beliebige Richtung ein.

Programmiertechnische Anforderungen

Auf den ersten Blick muten 2D-Spiele in ihrer Programmierung trivialer an als ihre 3D-Genossen – immerhin könnte man meinen, sie seien durch den Wegfall einer Dimension weniger komplex. Doch genau dieser Dimensionsbeschnitt wirft neue Fragen auf: Wann liegen Gegenstände oder Spielfiguren vor- oder hintereinander? Wann müssen sie sich ausweichen und wo dürfen die Charaktere stehen? Die fehlenden Tiefeninformationen müssen also auf anderem Weg simuliert werden. Die Engine von The Whispered World bietet deshalb die Möglichkeit, Hintergründe und Figuren aus mehreren Ebenen zusammenzusetzen und mit entsprechenden Informationen zu versehen.

Ein weiteres großes Problemfeld bei 2D-Adventures ist die Verarbeitung des von den Künstlern erschaffenen Bildmaterials. Da die Grafiken im Spiel genauso aussehen müssen, wie die Zeichner sie entworfen haben, kommen einige Anforderungen und Beschränkungen auf die Entwickler zu. Zum einen die pixelgenaue Darstellung der Bilder: Beim Parallax-Scrolling kommt es darauf an, dass die Bildebenen exakt übereinander liegen, um falsches Überlappen von Bildbereichen zu verhindern. Zum anderen können Texturen auch nicht hoch-skaliert werden, falls der Benutzer in einer höheren Auflösung spielen möchte – das würde entsprechend höher aufgelöstes Bildmaterial voraussetzen.

Texturverwaltung

Perspektivische Änderungen können in 3D-Spielen dynamisch zur Laufzeit mit Hilfe von Matrizen und Vektoren berechnet werden. Bei 2D-Spielen sieht das anders aus. Hier hat eine Änderung der Perspektive einen Wechsel der Textur zur Folge. Besonders interessant wird das bei Animationen, denn allein Sadwick hat bei acht Bewegungsrichtungen mit je zwölf Frames eine Summe von 96 Einzelbildern. Da heutige Grafikkarten weniger auf das schnelle Wechseln von Texturen optimiert sind, können solch scheinbar einfache Vorgänge sehr performance-lastig werden. Diese Pixelflut muss daher effizient verwaltet werden. Um das ständige Laden von Texturen in den Grafikkartenspeicher zu vermeiden, halten wir die Anzahl der Texturen im Hauptspeicher möglichst gering. Deshalb erzeugt The Whispered World Texturen, die die maximale Größe annehmen, die von der Grafikkarte des Spielers unterstützt wird. Diese meist 2048x2048 Pixel großen Texturen dienen als Container für alle Grafiken, die in einer Szene im Spiel verwendet werden. Ist eine solche Container-Textur erst einmal in den Buffer der Grafikkarte geladen, kann die Hardware schnell auf einzelne Texturbereiche zugreifen, die jeweils kleine Grafiken repräsentieren. Um möglichst wenige Textur-Container zu erzeugen, ordnen wir das Bildmaterial möglichst effizient und platzsparend auf dem Container an. Dafür kommt eine speziell geschriebene Applikation zum Einsatz, die im Pre-Processing-Bereich die Texturen unter maximaler Ausnutzung einer Container-Textur anordnet.

Gunnar Bergmann, Franziska Reinhard, Fabian Pahl

Problem erkannt, Problem gebannt

Portierung von Direct3D 9 zu Direct3D 10

Konnte man beim letzten Direct3D-Wechsel die notwendigen Anpassungen noch unkompliziert ausführen, erfordert Direct3D 10 einiges mehr an Aufwand. Dieser Artikel räumt die größten Hürden aus dem Weg.

Ralf Kornmann

ist Lead Software Engineer bei EA Phenomic.

Ralf Kornmann wurde von Microsoft seit 2006 dreimalig mit dem MVP-Award für den Bereich DirectX ausgezeichnet. Nachdem er bei Nonatainment für die Entwicklung von Direct3D-Tools zuständig war, übernahm er 2007 die Engine-Entwicklung bei EA Phenomic. Derzeit arbeitet er am Echtzeit-Strategiespiel Battleforge.

Bevor wir mit der eigentlichen Portierung beginnen, muss die vorhandene Direct3D-9-Engine auf Stolpersteine untersucht werden – vor allem, wenn die Engine auch nach dem Umbau noch Windows XP oder ältere Hardware unterstützen soll. Bei Direct3D 10 strich Microsoft alle vordefinierten Funktionen (Fixed Functions) im Vertex- und Pixelbereich ersatzlos, daher sollten Sie diese zuerst überarbeiten. Dazu gehören offensichtliche Methoden wie SetLight, SetMaterial oder SetTextureStageState, aber auch Render-States wie D3DRS_AMBIENT oder D3DRS_NORMALIZENORMALS. Auch zur Shader-Pipeline gehörende Methoden wie SetVertexShader und SetPixelShader müssen untersucht werden, da ein Aufruf mit einem NULL-Zeiger zur Deaktivierung des Shaders führt. Kommen FX-Dateien in nativer oder bereits kompilierter Form zum Einsatz, beziehen Sie auch diese in die Überprüfung mit ein.

Für alle so gefundenen Render-Operationen implementieren Sie nun eine Shader-Lösung. Dabei müssen Sie sich entscheiden, ob die Engine weiterhin nicht shaderfähige Hardware unterstützen soll oder eben nicht. Wenn Sie die Unterstützung streichen, entfernen Sie nach einem erfolgreichen Test der Shader-Lösung die alte Variante – das spart Code-Zeilen und vermeidet Fehlerquellen. Um bei der Entwicklung der Shader-Lösung nicht in eine weitere Falle zu laufen, sollten Sie zur Entwicklung unbedingt die Direct3D-Shader-Hochsprache HLSL (High Level

Das Strategiespiel **Battleforge wird Direct3D 10 unterstützen**, die Programmierer verwenden bei der Portierung die hier vorgestellten Techniken.

Shader Language) verwenden, denn Direct3D 10 unterstützt weder in Assembler programmierte Shader noch in bereits kompilierter Form vorliegende Shader älterer Direct3D-Versionen. Um ein neues Kompilieren kommen Sie daher nicht herum. Nach dem Ersetzen der Fixed Functions untersuchen Sie die Direct3D-9-Engine auch auf die Verwendung von in Assembler programmierten Shadern und wandeln diese gegebenenfalls in eine HLSL-Variante.

Im nächsten Schritt untersuchen Sie die Engine auf die Verwendung von regulärem Hauptspeicher als Datenquelle für einzelne Objekte. Diese zeigt sich durch die Verwendung der Methoden DrawIndexedPrimitiveUP sowie DrawPrimitiveUP. Da Direct3D 10 diese Funktionalität nicht mehr unterstützt, müssen dynamische Vertex- und Index-Buffer zum Einsatz kommen. Zu guter Letzt sollten Sie auf die Verwendung des alten Flexiblen Vertex Formats (FVF) zugunsten von Vertex-Deklarationen verzichten.

Erzeugung eines Direct3D-10-Devices

Neben Direct3D 10 enthält Vista mit DXGI (DirectX Graphics Infrastructure) noch eine weitere API. Diese übernimmt ab Direct3D 10 versionsunabhängige Aufgaben wie das Verwalten der verfügbaren Hardware und die Swapchain-Steuerung. Dementsprechend beginnt der Einstieg in Direct3D 10 auch bei der DXGI-Funktion CreateDXGIFactory, welche einen Snapshot der momentan verfügbaren Grafik-Hardware erzeugt. Im Gegensatz zu Direct3D 9 erfasst Direct3D 10 nicht die angeschlossenen Ausgabegeräte, sondern die tatsächlich vorhandenen GPUs (EnumAdapters). Diese erhält man nun durch wiederholte Aufrufe von EnumOutputs des entsprechenden IDXGI-Adapter-Interfaces. Da die IDXGI-Interfaces keine Auskunft über die Anzahl der vorhandenen Adapter und Ausgabegeräte geben, müssen beide Methoden so lange mit steigendem Index aufgerufen werden, bis sie anstelle eines S_OK den Fehlercode DXGI_ERROR_NOT_FOUND liefern. Sowohl IDXGI-Adapter wie auch IDXGI-Output unterstützen eine GetDesc-Methode, welche eine entsprechende Informationsstruktur ausfüllt. Daneben bietet der IDXGI-Adapter noch die Methode CheckInterfaceSupport an, mit der die korrekte Treiberversion für eine Direct3D-Version ermittelt wird. Machen Sie aber nicht den Fehler, vom Vorhandensein eines Treibers darauf zu schließen, dass der Adapter diesen auch nutzen kann. Der einzige zuverlässige Test ist das Erzeugen des Direct3D-Devices.

Dafür stellt Direct3D 10 zwei Funktionen zur Verfügung: Während D3D10CreateDevice lediglich das Device erzeugt, kreiert D3D10CreateDeviceAndSwapChain zusätzlich noch ein Swap Chain. In beiden Fällen ist es möglich, über die DXGI Factory nachträglich weitere Swap Chains zu kreieren. Zum Erzeugen eines Hardware-Devices benötigt Direct3D 10 den zuvor ermittelten Adapter. Optional kann hier auch NULL verwendet werden, in diesem Fall erzeugt Di-

```
void Render::SetCullMode (CullMode mode)
{
    DWORD D3D9Mode = ConvertMode (mode);

    if (m_States[D3DRS_CULLMODE] != D3D9Mode)
        m_pDevice->SetRenderState (D3DRS_CULLMODE, D3D9Mode);
}
```

So sieht eine typische State-Management-Methode unter Direct3D 9 aus.

```
void Render::SetCullMode (CullMode mode)
{
    D3D10_CULL_MODE D3D10Mode = ConvertMode (mode);

    if (m_RasterrizerDesc.CullMode == D3D10Mode)
        return;

    m_RasterrizerDesc.CullMode = D3D10Mode;

    ID3D10RasterizerState* pState;

    m_pDevice->CreateRasterizerState (m_RasterrizerDesc, &pState);

    m_pDevice->RSSetState (pState);

    m_pOldRasterizerState->Release ();
    m_pOldRasterizerState = pState;
}
```

Eine naive Portierung von D3D9 zu D3D10 – die ständigen CreateRasterizerState-Aufrufe bei jeder Änderung verursachen jedoch einen großen Overhead.

```
void Render::SetCullMode (CullMode mode)
{
    D3D10_CULL_MODE D3D10Mode = ConvertMode (mode);

    if (m_RasterrizerDesc.CullMode == D3D10Mode)
        return;

    m_RasterrizerDesc.CullMode = D3D10Mode;
    m_RasterizerChanged = true;
}

void Render::UpdateStates ()
{
    if (m_RasterizerChanged)
    {
        ID3D10RasterizerState* pState;

        m_pDevice->CreateRasterizerState (m_RasterrizerDesc, &pState);

        m_pDevice->RSSetState (pState);

        m_pOldRasterizerState->Release ();
        m_pOldRasterizerState = pState;

        m_RasterizerChanged = false;
    }

    //... update the other state objects
}

void Render::Draw (...)
{
    UpdateStates ();

    //... Draw call
}
```

Auch wenn Sie die State Objects erst zum Draw Call erzeugen, bleibt das Problem der übermäßig hohen Anzahl von CreateRasterizerState.

```
void Render::SetCullMode (CullMode mode)
{
    if (m_RasterrizerDesc.CullMode == mode)
        return;

    m_RasterrizerDesc.CullMode = mode;
    m_RasterizerChanged = true;
}

ID3D10RasterizerState* GetRasterizerState ()
{
   map <RasterrizerDesc, ID3D10RasterizerState*> :: const_iterator i;

   i = m_RasterrizerStateStore.find( m_RasterrizerDesc );

   if ( i != m_RasterrizerStateStore.end( ) )
       return i->second;

   D3D10_RASTERIZER_DESC Desc10;

   ConvertDesc (m_RasterrizerDesc, Desc10);

   ID3D10RasterizerState* pState;

    m_pDevice->CreateRasterizerState (Desc10, &pState);

    m_RasterrizerStateStore.insert (pair <RasterrizerDesc, ID3D10RasterizerState*> (m_RasterrizerDesc,
    pState));

    return pState;
}
void Render::UpdateStates ()
{
    if (m_RasterizerChanged)
    {
        ID3D10RasterizerState* pState = GetRasterizerState ();

        m_pDevice->RSSetState (pState);

        m_RasterizerChanged = false
    }

    //... update the other state objects
}
void Render::Draw (...)
{
    UpdateStates ();

    //... Draw call
}
```

Eine verbesserte Portierung speichert die einmal erzeugten State Objects und verwendet sie erneut.

```
void Render::SetRasterizerState (RasterizerState* pState)
{
    ID3D10RasterizerState* pD3D10State = (D3D10RasterizerState*)pState->GetD3D10State ();

    if (m_pCurrentRasterizerState == pD3D10State)
        return;

    m_pDevice->RSSetState (pD3D10State);
    m_pCurrentRasterizerState != pD3D10State
}
```

Im Optimalfall arbeitet das Interface des Renderers mit State Objects zusammen.

```
void Render::SetRasterizerState (RasterizerState* pState)
{
    D3D9RasterizerState* pD3D9State = (D3D9RasterizerState*)pState;

    if (m_pCurrentRasterizerState == pD3D9State)
        return;

    if (m_States[D3DRS_CULLMODE] != pD3D9State.m_CulldMode)
        m_pDevice->SetRenderState (D3DRS_CULLMODE, pD3D9State.m_CulldMode);

    \\ Update other states
}
```

Soll das Interface des Renderers mit State Objects zusammenarbeiten, müssen Sie den Direct3D-9-Renderer entsprechend anpassen.

Vor allem die vielen Shader-Effekte lassen sich unter **Direct3D 10 ressourcenschonender** umsetzen.

rect3D selbstständig eine DXGI-Factory und nutzt den ersten Adapter. Daneben werden – wie bei Direct3D 9 – auch nachladbare Software-Devices, ein NULL-Device sowie die bekannte Referenzimplementierung erzeugt, Letzte allerdings nur, wenn ein SDK installiert ist. Die Möglichkeit, ein Software-Device zu erzeugen, ist dagegen nur von theoretischem Interesse, da es derzeit keine entsprechende Implementierung gibt. Daher übergibt man als Modulhandle immer eine NULL. Über einen Flag-Parameter bestimmen Sie die zu aktivierenden API-Layer, dazu aber später mehr. Abschließend übergeben wir noch die SDK-Version, um dann ein ID3D10Device-Interface zu erhalten. Zudem erwartet D3D10CreateDeviceAndSwapChain eine vollständige Beschreibung des Swap Chains. Die Direct3D-10.1-Varianten dieser Funktionen entsprechen ihren Direct3D-10-Gegenstücken, lediglich ein weiterer Parameter wurde hinzugefügt. Über den Hardware-Level kann der Funktionsumfang des zu erzeugenden Devices bestimmt werden. Die Create-Funktion schlägt allerdings fehl, wenn Sie versuchen, einen Hardware-Level anzufordern, der von dem Adapter nicht unterstützt wird. Entsprechend kann mit einer Direct3D10.0-Hardware kein 10.1-Device erzeugt werden. Das Erzeugen eines 10.0-Devices bei einer 10.1-Hardware ist dagegen kein Problem, jedoch können die zusätzlichen Funktionen dann trotz eines ID3D10-Device1-Interfaces nicht genutzt werden. Da es keine Möglichkeit gibt, im Vorfeld den Funktionslevel eines Adapters zu bestimmen, müssen Sie dies durch das testweise Erzeugen entsprechender Devices erledigen.

API Layer

Direct3D 10 kennt vier verschiedene API-Layer, die aufeinander aufbauen. Basis von allem ist der Core-Layer. Er wird in jedem Fall erzeugt und übernimmt dabei die Kommunikation mit dem Gerätetreiber für den entsprechenden Adapter. Über dem Core-Layer kann ein Thread-Safe-Layer liegen, welcher verhindert, dass mehr als ein Thread gleichzeitig auf das Device zugreift. Im Gegensatz zu Direct3D 9, das standardmäßig keinen solchen Schutz bietet, können Sie diese Sicherheitsabfrage bei Direct3D 10 explizit mit dem D3D10_CREATE_DEVICE_SINGLETHREADED-Flag deaktivieren. Aus Performancegründen empfehle ich das aber nur für den Fall, wenn Direct3D 10 lediglich aus einem einzigen Thread benutzt wird.

Eine weitere Neuerung gegenüber Direct3D 9 ist das ID3D10Multithread-Interface, welches eine bessere Kontrolle des Multithread-Verhaltens des Devices erlaubt. So ist es mit den Methoden Enter und Leave nun möglich, eine Benutzung des Devices durch einen anderen Thread für mehrere Aufrufe zu unterbinden. Mit SetMultithreadProtected kann der Multithread-Schutz jederzeit zur Laufzeit aktiviert und auch wieder deaktiviert werden. Dies kann von Vorteil sein, wenn zeitweise mehr als ein Thread genutzt wird.

Der Debug-Layer ersetzt die bekannte Direct3D-9-Debug-Runtime, unterscheidet sich jedoch in einigen Punkten. Zunächst kann die Applikation durch das D3D10_CREATE_DEVICE_DEBUG-Flag das Erzeugen des Debug-Layer selbst veranlassen. Dies ist jedoch nur möglich, wenn das SDK auf dem Computer installiert ist. Entsprechend sollte dieses Flag nicht in Applikationen benutzt werden, welche auf Rechnern ohne SDK lauffähig sein müssen. Ein weiterer Unterschied besteht darin, dass der Debug-Layer nicht in das Verhalten von Direct3D 10 eingreift. Alle Aufrufe werden unverändert an den Core-Layer weitergereicht, auch wenn der Debug-Layer einen Fehler erkannt hat. Diese werden weiterhin als Debug-Output ausgegeben, können jedoch auch über das neue ID3D10InfoQueue-Interface vom Programm selbst weiterverarbeitet werden. Ebenfalls nur bei installiertem SDK ist der »Switch To Reference«-Layer verfügbar. Dieser erlaubt es, jederzeit zwischen dem eigentlichen Hardware-Device und dem Reference-Device umzuschalten.

Ressourcen

Eine der wesentlichen Änderungen bei Direct3D 10 finden Sie bei der Ressourcen-Verwaltung. Mussten Sie bei Direct3D 9 noch einen Speicherpool spezifizieren, wählt Direct3D 10 diesen eigenständig aufgrund der angegebenen Verwendung aus. Hierbei unterscheiden wir zwischen vier neuen Runtime-Klassen:

- **Immutable:** Ressourcen dieses Typs werden einmalig beim Erzeugen mit Daten gefüllt und können später nur noch von der GPU gelesen werden.
- **Default:** Der Standardtyp für alle Ressourcen, die von der GPU gelesen und beschrieben werden sollen.
- **Dynamic:** Hierbei handelt es sich um Ressourcen, die von der CPU beschrieben und der GPU gelesen werden können. Diesen Typ sollten Sie jedoch nur dann verwenden, wenn Sie die enthaltenen Daten einmal oder öfter pro Frame ändern.
- **Staging:** Bei Staging-Ressourcen handelt es sich um Ressourcen, die von der CPU gelesen oder beschrieben werden können. Sie dienen vordringlich als Ziel und Quelle für Kopieroperationen mit Default-Ressourcen.

Aufgrund des fehlenden Speicherpools kennt Direct3D 10 auch keinen Zwang mehr, Ressourcen, die direkt im Videospeicher angelegt sind, bei einem Reset neu zu erzeugen. Solange es nicht zu einer vollständigen Entfernung des Devices kommt, stellt die Runtime sicher, dass alle Daten erhalten bleiben. Dies hat jedoch auch Auswirkungen auf den Zugriff. Erlaubte Direct3D 9 noch das direkte Lesen und Schreiben einer Managed-Ressource, ist dies bei Direct3D 10 nicht mehr möglich. Um den Inhalt einer Default-Ressource zu ändern, muss diese entweder per UpdateSubressource mit Hauptspeicher-Da-

Eine typische Direct3D-9-Methode zum Aktualisieren der Shader-Konstanten.

```
void Render::SetPixelConst (int slot, const float* pValues, int count)
{
    m_pDevice->SetPixelShaderConstantF(slot, pValues, count);
}
```

Der falsche Weg: Ein ständiges Aktualisieren des Constbuffers erzeugt ein hohes Datenvolumen.

```
void Render::SetPixelConst (int slot, const float* pValues, int count)
{
    memcpy (&m_PixelConstValues[slot*4], pValues, count*4*sizeof(float);

    void* pData;

    m_pPixelConstBuffer->Map (D3D10_MAP_WRITE_DISCARD, 0, &pData);
    memcpy (pData, m_PixelConstValues, sizeof(m_PixelConstValues));
    m_pPixelConstBuffer->Unmap ();
}
```

Eine bessere Variante sammelt die Änderungen bis zum nächsten Draw Call.

```
void Render::SetPixelConst (int slot, const float* pValues, int count)
{
    memcpy (&m_PixelConstValues[slot*4], pValues, count*4*sizeof(float);
}

void Render::UpdateConsts ()
{
    void* pData;

    m_pPixelConstBuffer->Map (D3D10_MAP_WRITE_DISCARD, 0, &pData);
    memcpy (pData, m_PixelConstValues, sizeof(m_PixelConstValues));
    m_pPixelConstBuffer->Unmap ();
}

void Render::Draw ()
{
    UpdateConsts ();

    //... Draw
}
```

Dies kann weiter verbessert werden, indem die Obergrenze berücksichtigt wird.

```
void Render::SetPixelConst (int slot, const float* pValues, int count)
{
    memcpy (&m_PixelConstValues[slot*4], pValues, count*4*sizeof(float);
    m_LastPixelSlot = max (m_LastPixelSlot, slot+count);
}

void Render::UpdateConsts ()
{
    if (m_LastPixelSlot == -1)
        return;

    void* pData;

    m_pPixelConstBuffer->Map (D3D10_MAP_WRITE_DISCARD, 0, &pData);
    memcpy (pData, m_PixelConstValues, m_LastPixelSlot*sizeof(float)*4);
    m_pPixelConstBuffer->Unmap ();

    m_LastPixelSlot = -1;
}

void Render::Draw ()
{
    UpdateConsts ();

    //... Draw
}
```

Im Idealfall passen Sie die Größe des verwendeten Constbuffers an, da auch ungenutzte Bytes übertragen werden.

```
void Render::SetPixelConst (int slot, const float* pValues, int count)
{
    memcpy (&m_PixelConstValues[slot*4], pValues, count*4*sizeof(float);
    m_LastPixelSlot = max (m_LastPixelSlot, slot+count);
}

void Render::UpdateConsts ()
{
    if (m_LastPixelSlot == -1)
        return;

    ID3D10Buffer* pBuffer = GetPixelBuffer (m_LastPixelSlot);

    if (m_pCurrentPixelBuffer == pBuffer)
    {
        m_pDevice->PSSetConstantBuffer (m_PixelPerObjectSlot,1, &pBuffer);
        m_pCurrentPixelBuffer = pBuffer;
    }

    void* pData;

    pBuffer->Map (D3D10_MAP_WRITE_DISCARD, 0, &pData);
    memcpy (pData, m_PixelConstValues, m_LastPixelSlot*sizeof(float)*4);
    pBuffer->Unmap ();

    m_LastPixelSlot = -1;
}

void Render::Draw ()
{
    UpdateConsts ();

    //... Draw
}
```

Durch eine Änderung am Interface kann auch der memcpy beseitigt werden.

```
void* Render::BeginPixelConstUpdate (int slots)
{
    ID3D10Buffer* pBuffer = GetPixelBuffer (slots);

    if (m_pCurrentPixelBuffer == pBuffer)
    {
        m_pDevice->PSSetConstantBuffer (m_PixelPerObjectSlot,1, &pBuffer);
        m_pCurrentPixelBuffer = pBuffer;
    }

    void* pData;

    pBuffer->Map (D3D10_MAP_WRITE_DISCARD, 0, &pData);

    return pData;
}

void* Render::EndPixelConstUpdate ()
{
    m_pCurrentPixelBuffer->Unmap ();
}
```

Auch hier müssen Sie unter Umständen wieder die noch vorhandene Direct3D-9-Methode anpassen.

```
void* Render::BeginPixelConstUpdate (int slots)
{
    m_PixelConstSlots = slots;
    return m_pPixelConstBuffer;
}

void* Render::EndPixelConstUpdate ()
{
    m_pDevice->SetPixelShaderConstantF(0, m_pPixelConstBuffer, m_PixelConstSlots*sizeof(float)*4);
}
```

Battleforge mit Direct3D 9 oder 10 – optisch unterscheiden sich die Versionen kaum, **es soll mit D3D 10 aber schneller laufen.**

ten überschrieben werden oder von einer Staging-Ressource kopiert werden. Beim Auslesen ist eine Kopie in eine Staging-Ressource sogar unvermeidlich. Daneben erwartet Direct3D 10 noch eine Angabe, welche Einheiten den GPU-Zugriff auf die Ressource erhalten sollen. Im Vergleich zu Direct3D 9 haben Sie zwar größere Freiheiten, jedoch sollten Sie aus Leistungsgründen so wenige Einheiten wie möglich angeben.

Beim Portieren fällt unweigerlich auf, dass Direct3D 10 keine Surface-Klassen mehr kennt. Deren Aufgaben werden nun vordringlich durch reguläre 2D-Texturen übernommen. So werden nun Render Targets oder Depth Stencil Buffers als 2D-Textur mit entsprechender Bindung erzeugt. Die Verwendung von Surfaces zur Repräsentation von Mipmaps wurde dagegen ersatzlos gestrichen. Alle Operationen, die sich auch auf Mipmaps beziehen können, akzeptieren nun eine Subressource-ID.

Dies gilt auch für die Definition eines Views. Views stellen ein neues Konzept in Direct3D 10 dar, bei dem durch ein zusätzliches Objekt beschrieben wird, auf welche Teile einer Ressource die GPU zugreifen kann. Hierfür gibt es drei Arten von Views: Shader Ressource Views kommen als Datenquellen für alle Shader zum Einsatz, Render Target und Depth Stencil Views definieren dagegen, welche Daten als Depth Stencil und Color Buffer dienen sollen. Um einen View erfolgreich zu erzeugen, muss die entsprechende Ressource für die entsprechende Verwendung geflaggt sein.

Shader

Obwohl der Shader-Compiler für Direct3D 10 dank eines Kompatibilitätsmodus auch für Direct3D 9 programmierte Shader kompilieren kann, ist diese Möglichkeit mit Vorsicht zu genießen, denn in diesem Modus werden alle im Code aufgeführten globalen Variablen in einem einzigen konstanten Puffer zusammengefasst. Dadurch tritt folgendes Problem auf: Der Puffer kann nun große unbenutzte Bereiche enthalten. Da Konstanten-Puffer jedoch immer vollständig aktualisiert werden, entsteht ein sehr hohes Datenvolumen zwischen CPU und GPU. Dies kann sogar zu einem schwer zu entdeckenden Flaschenhals werden und so die Leistung erheblich in den Keller drücken.

Um diesen Engpass zu verhindern, sollten Sie alle globalen Variablen abhängig von der Änderungsrate innerhalb des HLSL-Codes einem Konstanten-Puffer zuordnen. In der Praxis haben sich dabei drei Puffer als gute Ausgangsbasis herausgestellt: einer für Parameter, die sich selten oder nie während des Programmablaufes ändern, ein weiterer für alle Parameter, die sich pro Frame verändern können, sowie einen für Objekt-Änderungen. Während der letzte Puffer jeweils für den Shader spezialisiert werden sollte, empfiehlt es sich, die beiden anderen immer zu verwenden. Können Sie diese ideale Lösung nicht umsetzen, sollten Sie jedoch unbedingt die Anzahl der unbenutzten Elemente im Konstanten-Puffer reduzieren. Der Shader-Disassembler kann dabei eine große Hilfe sein.

Statemanagement

Während Direct3D 9 auf eine Menge von einzeln veränderbaren Renderstates setzt, nutzt Direct3D 10 sogenannte State Objects (eine Sammlung von funktional zusammengehörigen States). Genau wie alle anderen Direct3D-Ob-

jekte werden sie mit entsprechenden Create-Methoden erzeugt. Danach sind die enthaltenen Werte für die einzelnen States jedoch nicht mehr änderbar. Im Gegensatz zu den Shader-Objekten kann hier jedoch der gespeicherte Zustand mit der Funktion GetDesc abgefragt werden. In besonderen Fällen kann dieser von den ursprünglich zur Erzeugung benutzten Werten abweichen. Direct3D 10 setzt nicht aktive States auf ihren Standardwert, beispielsweise bei Blend-Operationen eines nicht aktiven Alphablendings.

Eine weitere Eigenheit von Direct3D-10-State Objects liegt darin, dass es keine zwei Objekte mit den gleichen Zustandswerten gibt. In solchen Fällen erzeugt die entsprechende Create-Methode kein neues Objekt, sondern liefert das bereits vorhandene mit einem erhöhten Referenzcounter zurück. Auf diese Weise kann auch die Limitierung auf maximal 4.096 State Objects pro Typ leicht eingehalten werden.

Die neuen State Objects erweisen sich bei der Portierung leicht als Stolperstein, wenn überlagerte Ebenen auf einer Einzel-State-Schnittstelle bestehen. Entsprechend wäre eine Anpassung dieser höheren Layer sinnvoll. Dies erfordert dann jedoch auch, dass bei einer weiteren Nutzung von Direct3D 9 dort ebenso State Objects implementiert werden müssen. Entscheiden Sie sich gegen diesen Weg, müssen Sie für Direct3D 10 eine Einzel-State-Verwaltung entwickeln. Ein naiver Ansatz dafür wäre, jeweils ein State Object pro Typ zu halten und es bei jeder Zustandsänderung zu verwerfen und ein neues zu erzeugen. Eine scheinbar bessere Lösung würde dies nur als Teil der Draw-Operation tun. In beiden Fällen würde es aber zu unzähligen Aufrufen von Create-Methoden kommen – diese kosten auch bei Direct3D 10 nach wie vor sehr viel Leistung. Daher sollte eine Einzel-State-Verwaltung bereits erzeugte Kombinationen speichern und bei Zustandsänderungen die richtige auswählen und aktivieren.

Vertex Deklarationen / Input Layouts

Direct3D 10 ersetzt die aus Direct3D 9 bekannten Vertex-Deklarationen durch ein neues Objekt. Im Gegensatz zur bisherigen Vertex-Deklaration erfolgt die Bindung zwischen dem Vertex-Layout und den Shader-Registern nun nicht mehr zur Laufzeit, sondern direkt beim Erzeugen. Aus diesem Grund benötigt die entsprechende Create-Methode neben der Beschreibung des Vertex-Layouts auch noch die Input-Signatur des Vertex-Shaders, mit dem die Daten später verarbeitet werden sollen. Diese Signatur ist im Binärcode des Shaders enthalten, und so können Sie der Einfachheit halber die gleichen Daten benutzen, die auch zum Erzeugen des Shaders selbst zum Einsatz kommen.

Bei einer großen Anzahl von Vertex-Shadern steigt jedoch zwangsläufig auch die Anzahl der Input-Layouts. Um dies zu verhindern, können unter Direct3D 10 Shader mit gleicher Input-Signatur das gleiche Input-Layout nutzen – dazu muss aber die Signatur von jedem Shader ermittelt werden. Ähnlich wie bei den State Objects gilt auch bei den Input-Layouts, dass sie möglichst außerhalb des eigentlichen Render-Loops erzeugt werden sollten. Ist dies nicht möglich, sollten Sie auf eine Cachestruktur setzen, um zu verhindern, dass die Input-Layouts in jedem Frame erneut erzeugt werden müssen.

Ausblick

Bereits heute steht fest, das Direct3D 11 in naher Zukunft Direct3D 10 ablöst. In Anbetracht der unzähligen Hürden bei der Portierung entsteht schnell die Überlegung, auf Direct3D 11 zu warten. Die Herausforderung wird dabei jedoch nicht kleiner werden, denn Direct3D 11 stellt lediglich eine Erweiterung von Direct3D 10 dar. Auch bei der Kompatibilität verzichtet Microsoft auf einen erneuten harten Schnitt. Sowohl Windows Vista wie auch jede Direct3D-10-Hardware werden unterstützt. Entsprechend ist eine Portierung von Direct3D 10 zu 11 eine Aufgabe von wenigen Stunden. Anschließend können die neuen Möglichkeiten wie das verbesserte Multithreading, Compute Shader oder Tesselation verwendet werden.

Ralf Kornmann

GAMEINDUSTRYJOBS.DE
The ultimate job within two seconds
POST A JOB
Search
Post your job at no charge in 2008

Teil 1: Die Form. Wer soll das alles lesen?

So funktioniert Designdokumentation

Serie

So funktioniert Designdokumentation

- Die Form: Wer soll das alles lesen?
- Der Inhalt: Habe ich etwas vergessen?

Das so genannte DDD, das Detailed Design Document, ist die Bibel der Spieleentwickler. Zumindest in der Theorie. In der Praxis wird das DDD schon mal als »Staubfänger« verspottet. Ein vermeidbares Schicksal.

Dirk Riegert

ist Lead Game Designer bei Related Designs.

Dirk ist seit 2002 Game Designer bei Related Designs und hat dort unter anderem an den beiden RTS-Titeln No Mans Land (CDV) und Castle Strike (Data Becker) mitgewirkt. Von 2004 bis 2006 war er als Lead Game Designer für die Konzeption und Kommunikation der spielerischen Inhalte von Anno 1701 (Sunflowers) verantwortlich. Zur Zeit arbeitet er als Lead Designer an Anno 1404.
riegert@related-designs.de

Die schlechte Nachricht gleich zu Beginn: Game Designer haben es schwer! Es gibt (noch?) keine Sprache, kein Notationssystem, mit dem sie die Regeln eines Spiels allgemeingültig erfassen können. Wo Komponisten ihre Sinfonien auf Notenpapier für jeden Orchestermusiker verständlich festhalten, sind Game Designer (oft am Rande des Nervenzusammenbruchs) emsig damit beschäftigt, kryptische Konzepte auf Notizzettel zu kritzeln oder liebevolle Designerprosa in Telefonbuchstärke zu erstellen. Der Lohn solcher Mühen ist nicht selten ein Kommentar der Marke »Wer soll das alles lesen?«; ein ebenso einleuchtender wie niederschmetternder Einwurf.

Kenne deine Leser!

Damit einem traurige Erlebnisse dieser Art erspart bleiben, sollte man sich als Verfasser von Designdokumenten zunächst über die wichtigste Frage klar werden: Wer wird das Designdokument lesen? Unterschiedliche Leser bedeuten unterschiedliche Anforderungen an ein DDD:

- **Designer:** Sie wollen Ideen festhalten und miteinander austauschen.
- **Publisher:** Er will – meist in Gestalt des Producers – überprüfen, ob Design und Produkt übereinstimmen und den Anforderungen gerecht werden.
- **Programmierer:** Sie sehen das DDD als Sammlung von Arbeitsaufträgen und Inhalten, die am Ende der Produktion ein fertiges Spiel ergeben sollen.
- **QA und Testing:** Sie wollen alle Regeln und Inhalte des Spiels überblicken und überprüfen, ob sie die erforderliche Qualität und Quantität aufweisen.
- **Outsourcing Partner:** Sie wollen schnellen Zugriff auf alle für sie relevanten Infos zur Erstellung zusätzlicher Spielinhalte (z.B. Grafiken, Missionen, Musik etc.).

Das sind eine Menge unterschiedlicher Bedürfnisse und Interessen, die ein einzelnes zentrales DDD kaum stillen kann. Es mag daher sinnvoll sein, spezielle Dokumente neben dem eigent-

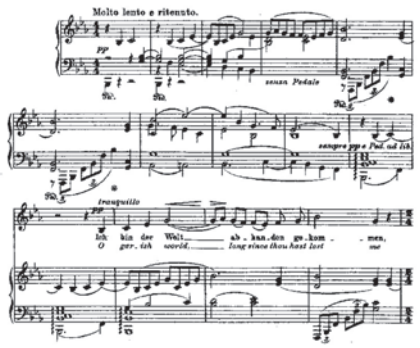
Was Designer wissen sollten

Dinge, die Programmierer mögen:

- Hauptsätze
- Bulletpoints
- Imperativ
- Diagramme

Dinge, die Programmierer hassen:

- Nebensätze
- Fließtext
- Konjunktiv
- Designerprosa

Was macht Anno aus? Ein Beispiel für den Einsatz einfacher und **unmissverständlicher Symbolik.**

Komponisten haben ihre eigene **universelle Zeichensprache**. Davon können Gamedesigner nur träumen.

Pre-Alpha-Screenshot einer **orientalischen Hafeninsel** in Anno 1404.

lichen DDD anzufertigen. Doch Vorsicht: Mehrere Dokumente bedeuten immer auch die Gefahr, dass nicht alle ausreichend aktualisiert und gepflegt werden. Nichts ist älter als die Zeitung von gestern? Wer immer dieses Sprichwort erfunden hat, kannte offensichtlich verwaiste Designdokumente nicht.

Im Zweifelsfall sollte man sich bei der Erstellung des DDD an den Bedürfnissen der Programmierer orientieren (siehe Kasten: Was Designer wissen sollten). Sie sind die größte und wichtigste Adressatengruppe der Game Designer; die Berücksichtigung ihrer Bedürfnisse ist für ein vitales Designdokument überlebenswichtig. Es empfiehlt sich also, die Programmierung bei der Strukturierung und Gestaltung des DDD eng einzubinden. So minimiert man die Gefahr, an denen vorbei zu formulieren, die dem Spiel Leben einhauchen.

Acht Faustregeln

Die folgenden Regeln helfen dabei, ein Designdokument zu erstellen, das nicht als Briefbeschwerer endet.

Ein nützliches DDD muss ...

- ... sich kurz fassen.
- ... sein Layout vereinheitlichen.
- ... Redundanzen vermeiden.
- ... Begründungen von Regeln trennen.
- ... Bilder, Diagramme und Flowcharts bevorzugen.
- ... Imperativ statt Konjunktiv nutzen.
- ... ein verbindliches Glossar enthalten.
- ... ständig aktualisiert werden.

Regel 1: Kurz fassen. Die wichtigste Regel. So banal sie klingt, so schwer ist es, sie zu befolgen. Designer neigen meist dazu, sich umständlich oder unverbindlich auszudrücken. Je komplexer das Thema, desto verworrener oft die Sprache. Es empfiehlt sich daher (gerade zu Beginn der Dokumentation), mit den Adressaten eines Eintrags diesen wiederholt zu besprechen und ihn (und die folgenden) anhand der aufkommenden Rückfragen zu optimieren.

Regel 2: Layout vereinheitlichen. Ein Designdokument benötigt einheitliche Strukturen. Gerade wenn mehrere Designer gemeinsam an einem großen Dokument arbeiten, läuft man Gefahr, Leser durch wechselnde Formatierungen und individuelle Schreibstile zu verwirren. Gemeinsam erarbeitete Formatvorlagen, die hin und wieder auf ihre Sinnhaftigkeit geprüft werden, wirken Wunder. Sie erfordern zwar die Selbstdisziplin aller Beteiligten, erleichtern aber die Lektüre ungemein.

Regel 3: Redundanz vermeiden. Wiederholungen sind der Feind des Designdokuments. Sie minimieren die Lesbarkeit und lenken vom Wesentlichen ab. Hinzu kommt: Was an zwei Stellen genannt wird, muss auch an zwei Stellen aktualisiert werden. Eine klare Priorisierung von Designinhalten und eine eindeutige Zuordnung zu einzelnen Features helfen, Redundanzen vorzubeugen.

Regel 4: Begründungen von Regeln trennen. Es kann wichtig sein, zu begründen, weshalb eine bestimmte Designentscheidung getroffen oder verworfen wurde. Diese Begründungen sollten jedoch nicht mit den Regeln eines Features vermischt werden, um Verwirrung zu vermeiden. Für unsere Anno-DDDs verwenden wir dazu beispielsweise FAQs, die am Ende jedes Designeintrags die wichtigsten Begründungen, Alternativen oder Problemstellungen mit einer Frage und einer dazugehörigen Antwort erfassen (siehe Kasten: Anno Standarddesigneintrag).

Regel 5: Bilder sagen mehr als Worte. Fließtext ist immer die schlechteste Lösung. Ist er nicht zu vermeiden, unbedingt auf übersichtliche Formatierungen achten! Ansonsten empfiehlt es sich, bei jeder Gelegenheit auf bessere Instrumente zurückzugreifen, seien es Listen, Tabellen, Diagramme oder Illustrationen. Sie können schneller erfasst werden und helfen dabei, komplexe Inhalte anschaulich darzustellen.

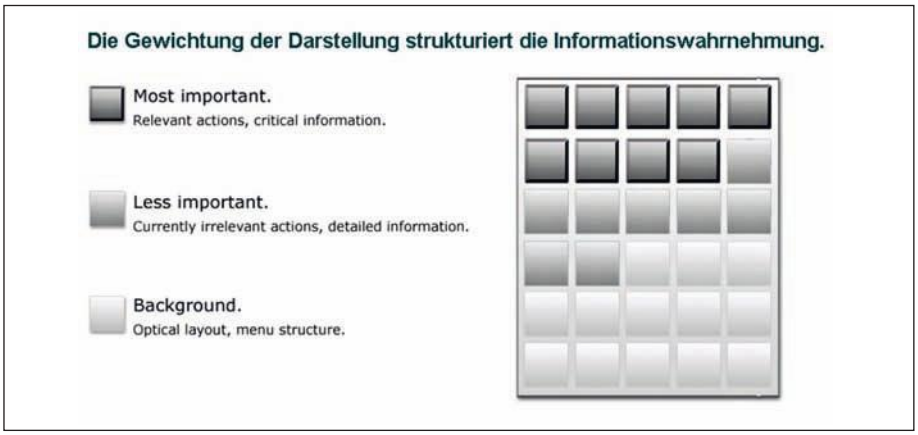
Regel 6: Imperativ statt Konjunktiv. Der Konjunktiv hat in einem Regelwerk nichts verloren.

Anno Standarddesigneintrag:

Hier der Aufbau eines typischen Anno-Feature-Eintrags, auf den sich das Gamedesign in Absprache mit der Programmierung geeinigt hat:

- **Name:** Name des jeweiligen Features.
- **Verantwortliche:** Jeder Eintrag hat einen verantwortlichen Designer und einen oder mehrere Bearbeiter, die für die Umsetzung zuständig sind. So weiß man immer, wer für Dokumentation, Implementierung und Aktualisierung geradesteht.
- **Status:** In der Kopfleiste eines Designeintrags kennzeichnet ein Statusdiagramm den Zustand jedes Designeintrags (erster Entwurf, vorläufig abgeschlossen, final etc.).
- **Definition:** Sie legt in einem Satz die Bedeutung eines Features fest.
- **Kurzbeschreibung:** Sie erfasst prägnant (in Bulletpoints) die Kernelemente des Features.
- **Schaubilder:** Flowcharts, Illustrationen, Mockups oder andere Hilfsmittel bieten eine möglichst anschauliche Übersicht über das Feature.
- **Regeln:** Alle dem Feature zugehörigen Regeln werden hier in knappen Sätzen klar beschrieben. Konkrete Inhalte, die häufigen Aktualisierungen unterliegen (z.B. einzelne Quests), werden vom Regelwerk getrennt dokumentiert.
- **Balancing:** Auflistung aller Werte, die später durch das Gamedesign manipuliert werden. Konkrete Balancingwerte werden nicht erfasst.
- **FAQ:** Begründungen zu speziellen Designentscheidungen in Frage-und-Antwort-Form.
- **Implementierungsdetails:** Ergänzende Anmerkungen der jeweiligen Sachbearbeiter, die das Feature ins Spiel implementieren.

Konzeptzeichnung von Al Zahir, dem Großwesir und Berater des Spielers. Artwork: Tobias Mannewitz.

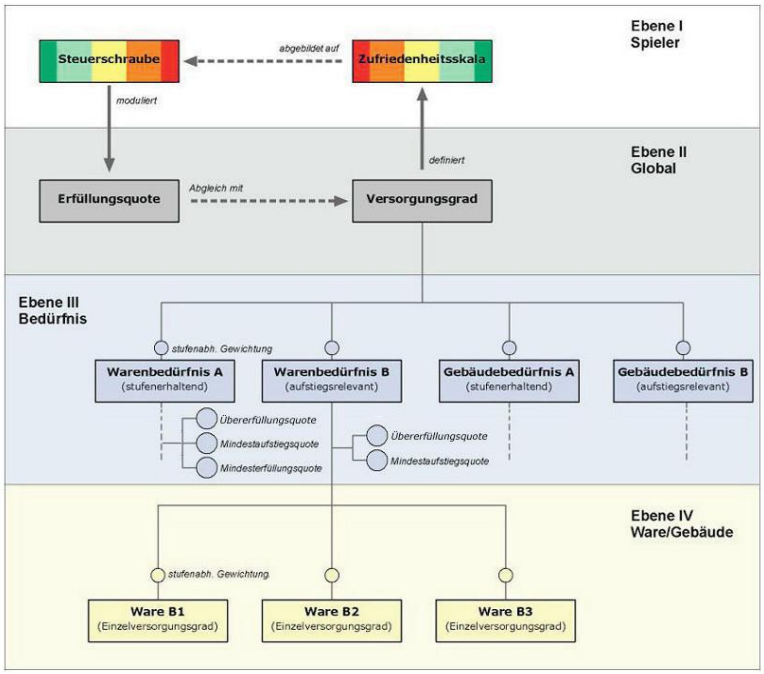


Übersichtliche Illustrationen helfen in Designdokumenten dabei, Botschaften für jeden intuitiv verständlich zu machen. Hier illustriert ein Schaubild die **Informationsgewichtung im Spielinterface**.

Er unterminiert die Aussagekraft des Designs. Die Verbindlichkeit von Regeln, die mit »sollte, könnte, würde« oder »hätte, wäre, wenn« garniert sind, geht gegen Null. Auch Wörter wie »eventuell« und »vielleicht« haben in einem DDD nichts verloren. Das Design wirkt auf Leser sonst unsicher und austauschbar. Stattdessen: Imperativ verwenden! (Und nicht: »Stattdessen könnte man auch den Imperativ verwenden.«)

Regel 7: Das Glossar nicht vergessen. Jedes Designdokument benötigt ein aktuelles und dem Team vertrautes Glossar. Zwei Namen für ein und dieselbe Sache sind zwar allzu menschlich, aber auch die natürlichen Feinde reibungsloser Kommunikation (Beispiel: »Ist die Grafik vom fliegenden Schloss schon geliefert worden?« »Keine Ahnung, aber gestern ist das schwebende Schloss rein gekommen.« »Hm... ist das vielleicht nicht dasselbe?«).

Übersicht Bedürfnissystem



Komplexe Sachverhalte können mittels Diagrammen und Flowcharts besser vermittelt werden.

Regel 8: Ständige Aktualisierung. Ein Designdokument, das nicht fortlaufend aktualisiert wird, wird unweigerlich nutzlos. Auch wenn es im fortschreitenden Produktionsprozess einen hohen Kraftaufwand bedeutet, ist es unerlässlich, dass das DDD stets die aktuellste Informationsquelle bleibt und nicht vom Flurfunk abgelöst wird. Im Fall des Scheiterns wird das DDD ein Schattendasein als ungeliebtes Mauerblümchen fristen und hilflos dem Chaos beiwohnen, das unweigerlich beginnt sich auszubreiten. Dieser letzte Satz war übrigens schlecht strukturiert und stilistisch schauderhaft – mit Absicht. Er ist ein typisches Beispiel ausufernder Designerprosa. Für Programmierer wäre ein knappes (aktuelles DDD = Ordnung) > (veraltetes DDD = Chaos) besser gewesen.

Word, Wiki oder was?

Welche Hilfsmittel man einsetzt, um Design zu dokumentieren, hängt vom jeweiligen Projekt und den Leuten ab, die das Dokument lesen und bearbeiten sollen. Für Anno haben wir uns zum Beispiel frühzeitig für eine Designdokumentation mittels eines firmeninternen, selbstentwickelten Intranets (Wiki) entschlossen. Im Laufe der Jahre hat sich diese Entscheidung insbesondere wegen der Komplexität und des Umfangs der Anno-Reihe sowie der hohen Akzeptanz durch das Team als richtig erwiesen. Middleware kann aber ebenso gut funktionieren.

Die größten Vorteile des Intranets sind für uns, dass ...

- ... übersichtliche Layouts leicht zu erstellen und zu verwalten sind.
- ... Einträge miteinander verlinkt werden können.
- ... jede Version eines Designeintrags jederzeit wieder hergestellt werden kann.
- ... alle Versionen eines Eintrags miteinander verglichen werden können.
- ... Einträge abonniert werden können, sodass man bei Aktualisierungen per Mail benachrichtigt wird.

Pre-Alpha-Screenshot einer ANNO-1404-Metropole mit einem im Bau befindlichen **Kaiserdom**, eine der wichtigsten Neuerungen im Spiel.

- ... Einträge wie in Foren von jedem Nutzer kommentiert werden können.
- ... Bilder, Galerien, Musik und Filme leicht zu integrieren sind.

Der größte Nachteil des Wikis ist seine Tendenz zu wuchern und dadurch zu einem undurchdringlichen Dickicht zu werden. Diese Eigenschaft hat das Intranet von seinem großen Bruder, dem Internet, geerbt. Nur die Selbstdisziplin der Nutzer und die weiter oben angesprochene Vereinheitlichung des Layouts können diesem schleichenden Drang Einhalt gebieten. Eine erträgliche Pflicht gemessen an den Vorteilen.

Natürlich können Designdokumente auch auf jede andere erdenkliche Art und Weise verfasst werden, wenn sie die Bedürfnisse ihrer Leser abdecken und den grundlegenden Anforderungen an ein DDD gerecht werden. Ob mit Windows verwaltet oder auf Butterbrotpapier gemalt: Am Ende zählt nur, ob die Verfasser und die Leser des Designdokuments ihr DDD lesen und verstehen, hegen und pflegen. Ist dies der Fall, werden eine reibungslosere Produktion und ein besseres Spiel die Folge sein.

Dirk Riegert

Vorschau

Der zweite Teil der Serie zum Thema Designdokumentation beschäftigt sich im nächsten Making Games Magazin mit den Inhalten eines Designdokuments. Welche Infos werden in welcher Weise erfasst? Wie strukturiert man die Vielzahl an Daten so, dass das DDD nicht zu einem Irrgarten wird?

Beispiel für einen guten und einen schlechten DDD-Eintrag

So nicht:

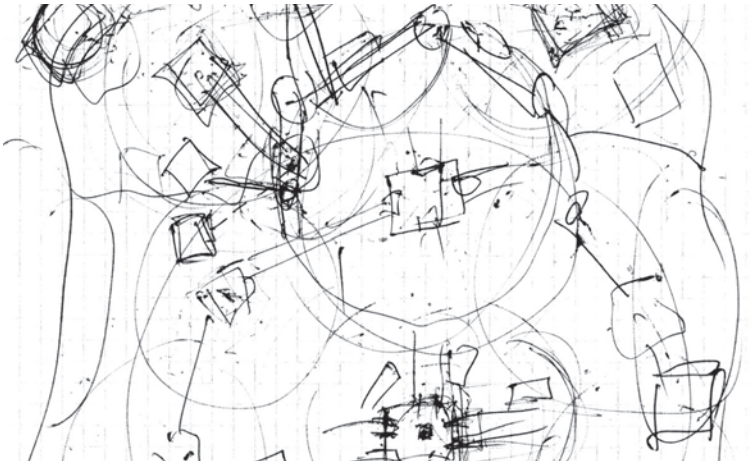
»Schiffe werden in verschiedene Schiffstypen unterteilt. Wir unterscheiden die stolzen Bezwinger der sieben Weltmeere am besten in Handelsschiffe, Kriegsschiffe und fliegende Schiffe. Handelsschiffe besäßen demnach keine Kanonen, könnten aber mehr Ladung an Bord nehmen als Kriegsschiffe und die fliegenden Schiffe. Die Kriegsschiffe sollten hingegen viel mehr Kanonen als fliegende Schiffe besitzen, könnten aber evtl. deutlich weniger Ladung an Bord nehmen als z.B. Handelsschiffe. Flugschiffe besäßen optimalerweise weniger Kanonen als militärische Schiffe, könnten dafür aber mehr Ladung an Bord nehmen als Letztere. Außerdem sollten Flugschiffe, wie schon ihr Name sagt, als einzige Schiffe hoch oben über den Wolken und Dächern der Städte ihre Runden drehen können.«

Warum schlecht?

Fließtext, keine Formatierung, uneinheitliche Namensgebung, Redundanzen, unklare Formulierungen und Designerprosa: Dieser Eintrag ist ein Alptraum für jeden Programmierer.

Bitte so:

Schiffstyp	Kanonen	Ladevolumen	Flugtauglichkeit
Handelsschiff	nein	[groß]	nein
Kriegsschiff	[viele]	[gering]	nein
Flugschiff	[wenige]	[mittel]	ja



Was aussieht wie unbeholfenes Gekritzel eines Fünfjährigen ist in Wirklichkeit das Ergebnis eines wilden **Brainstormings** zwischen Gamedesign und Programmierung.

Plattform-Debatte
Das Ende der Konsolen

Die aktuelle Konsolengeneration wird die letzte sein, sagt der DirectX-Erfinder Alex St. John. Ihr Geschäftsmodell hat sich überlebt, der PC weist mit Online-Vertrieb, Abo-Lösungen und Flexibilität den Weg in die Zukunft.

Alex St. John

ist CEO von WildTangent.

Alex St. John war die treibende Kraft hinter Microsofts DirectX-Architektur, die Windows in ein Betriebssystem für den Massenmarkt verwandelte und dem Markt für PC-Spiele und 3D-Hardware entscheidenten Anstoß gab. St. John fädelte einen Deal mit Sega ein, durch den Microsoft das Betriebssystem für die Dreamcast-Konsole auf Basis von DirectX entwickelte und so erstmals ins Konsolengeschäft vorstieß. Nach dem Fehlschlag der Dreamcast entschied sich Microsoft, selbst in den Konsolenmarkt einzusteigen und entwickelte die DirectXbox. Heute ist Alex St. John Geschäftsführer von WildTangent Inc., dem nach Yahoo und EA drittgrößten Spiele-Netzwerk in Nordamerika (weltweit: Nummer 5). WildTangent erreicht mehr als 50 Millionen Spieler im Monat.

Frühere Konsolengenerationen definierten sich durch wegweisende Grafik, hier etwa **Shenmue** (Dreamcast).

Wenn ich mir die aktuellen Entwicklungen im Spielemarkt ansehe, dann komme ich zu dem Schluss, dass die herkömmliche Spielkonsole vor dem Aussterben steht. Ich lehne mich sogar so weit aus dem Fenster zu sagen, dass die jetzige Generation von Konsolen die letzte in der Form sein wird, wie wir sie kennen. Das mag eine überraschende Position sein, wenn man bedenkt, dass wir uns mitten in einem Konsolenboom befinden. Aber ich werde im Laufe dieses Artikels auf klare und einfache Weise darlegen, warum das Sterben der Konsolen nicht nur unvermeidbar ist, sondern schon längst begonnen hat. Wir befinden uns auf dem Scheitelpunkt der Konsolenära. Der erste Schritt zum Verständnis dieses Trends ist es, nicht auf die Zahl der Spieler zu sehen, sondern auf das Geschäftsfeld. Es spielt keine Rolle, wie »populär« Konsolen sind. Ihr Problem ist, dass das zugrunde liegende Marktsystem kollabiert.

Die Ära der Grafik ist vorbei

Die Lebensfähigkeit des Konsolenmarkts hängt von mehreren wichtigen Faktoren ab. Erstens: Jede Konsolengeneration steht im Wettbewerb mit anderen Spieleplattformen und muss sich durch die Qualität und Vielfalt ihrer Inhalte bewähren. Neue Konsolen setzen dazu traditionell auf fortschrittliche Grafik. Sprunghafte Entwicklungen bei der 3D-Grafik sind das Kennzeichen der modernen Spiele-Ära. Der Übergang von einer Konsolengeneration zur nächsten erfolgt dann, wenn die Grafik der Vorgängerkonsole veraltet ist. Daraus folgt: Sobald sich eine Nachfolgekonsole nicht mehr durch grafische Fortschritte abheben kann, wird es diese Nachfolgekonsole nicht geben.

Das ist genau die Situation, die der Markt erreicht hat. Herausragende 3D-Grafik ist zu einer erschwinglichen Ware geworden. Die Ankunft von Direct3D auf dem PC hat einen Multimilliarden-Dollar-Markt für 3D-Hardware geschaffen, auf dem der Erfolg von Firmen wie ATI und Nvidia basiert. 3D-Grafikchips sind allgegenwärtig und günstig, und diesen Markt hat Microsoft absichtlich aus der Wiege gehoben.

»Indem Microsoft Low-level-Treiber definiert, die ein breites Sortiment von Hardware-Beschleunigern unterstützen und eine High-Level-Programmierschnittstelle darauf aufsetzt, erhält Microsoft enormen Einfluss auf die Richtung, die unabhängige Hardware-Hersteller bei ihren neuen Produkten einschlagen. Indem wir den Herstellern diktieren, welche Innovationen vom Betriebssystem unterstützt werden, welche Microsofts Weltbild entsprechen, welche sich verkaufen werden und welche nicht, bündeln wir den natürlichen Drang der Hersteller, sich voneinander abzusetzen, auf einem linearen, von uns definierten Weg. Wir zwingen Wettbewerber wie Apple dazu, mit unseren hochentwickelten Treibermodellen mitzuhalten, um am von uns neu geschaffenen Hardwaremarkt teilzuhaben. Wir geben Software-Herstellern Low-level-Zugang zu den Funktionen, die sie brauchen, behalten dabei aber die Kontrolle und knüpfen das Bedürfnis nach ›Echtzeit‹-Anwendungen an Windows.«

(Quelle: »Taking Fun Seriously II«, Strategiepapier von Microsoft zur Entwicklung von Direct3D)

Es ist für Konsolenhersteller wie Sony oder Nintendo mittlerweile unmöglich geworden, einen günstigeren, schnelleren 3D-Chip herzustellen als die, die der freie Markt hervorbringt. Microsoft und Nintendo haben dieses Problem erkannt und für ihre aktuellen Konsolen Grafikkomponenten von der Stange gewählt, in der Hoffnung, sich stattdessen durch Online-Funktionalität (Xbox 360) bzw. durch ein neues Eingabegerät (Wii) von der Konkurrenz abzuheben. Sony hat Milliarden in eine hauseigene 3D-Lösung investiert, ohne dadurch den geringsten Wettbewerbsvorteil zu erzielen. Erst mit Verspätung ging den Sony-Verantwortlichen auf, dass sie diesmal nicht mit Grafik punkten würden können, und sie setzten stattdessen auf die Blu-Ray-Unterstützung als Alleinstellungsmerkmal. Kurz gesagt: Die Ära, in der sich Konsolen allein durch ihre Grafikleistung voneinander unterscheiden, ist zu Ende.

Schutz vor Raubkopien wird unnötig

Der nächste wichtige Faktor, der die Überlebensfähigkeit von Konsolen bestimmt, ist das Verhältnis zwischen Konsolenherstellern und Spieleentwicklern bzw. -Publishern. Käufer halten ihre Konsolen für »Spiel-Ermöglichungsmaschinen«. Aus einem wirtschaftlichen Blickwinkel ist eine Konsole allerdings eine SpielVERHINDERUNGSmaschine. Eine Konsole ist ein riesiger, teurer Dongle, der seine Käufer daran hindert, Spiele zu spielen, für die sie nicht bezahlt haben. Diese Sicherheit ermutigt Entwickler dazu, zweistellige Millionenbeträge in ein Spiel zu investieren – in dem Wissen, dass es nur schwer raubkopiert werden kann. Aus diesem Grund führen sie 25 bis 30 Prozent ihres Umsatzes an den Konsolenhersteller ab. Diese Sicherheitsbeziehung macht das Konsolengeschäft für alle Beteiligten attraktiv. Was aber passiert, wenn community-basierte MMOGs wie World of Warcraft den Konsolenmarkt ebenso zu dominieren beginnen, wie sie es auf dem PC tun? World of Warcraft benötigt kein Konsolen-DRM, um sich vor Raubkopien zu schützen. Warum also sollte Blizzard seinen Umsatz mit einem Konsolenhersteller teilen? World of Warcraft gibt es aus offensichtlichen Geschäftsgründen nicht für die Konsole.

In einem Markt, in dem sich Spiele immer mehr durch ihre Community statt durch ihre Grafik hervorheben, hat die Konsole als Schutz vor Raubkopien nur noch eine geringe Bedeutung. Das zu verstehen ist wichtig, denn es hängt stark mit dem nächsten Punkt zusammen: nämlich, dass das Konsolengeschäft seit jeher ein Lockvogelgeschäft ist.

Ein Handelskanal bricht zusammen

Konsolen werden mit Verlust verkauft. Daran knüpft sich die Hoffnung, dass der Hersteller den Umsatzverlust durch Spieleverkäufe wettmachen kann. Der einzige Grund, warum Einzelhändler Konsolen verkaufen, ist die Erwartung, mit den Spielen Geld zu machen. Je mehr Spiele aber über das Internet vertrieben und an Online-Geschäftsmodelle wie Werbung oder Abogebühren geknüpft werden, desto stärker sinkt der Bedarf an klassischer Schachtelware – und entsprechend das Interesse daran, Konsolen im Einzelhandel zu verkaufen. Das ist genau das, was der Musikindustrie passiert ist, als sich das Geschäft ins Internet verlagerte. Ganze Regale voller CDs wurden durch ein paar iTune-Karten ersetzt, die an der Kasse hängen. Die einzige Rolle, die der Laden um die Ecke in der Online-Welt noch spielt, ist die als sicherer Austauschplatz von echtem gegen virtuelles Geld. In dem Maße, in dem die Konsolen ihr Geschäft in Richtung online verlagern, zersetzen sie gleichzeitig ihren wichtigsten Absatzkanal, den Einzelhandel. In ein paar Jahren wird vom Regalplatz für Spiele nichts anderes übrig sein als ein paar Haken, an denen Punktekarten für virtuelle Währungen baumeln.

Verluste jenseits von Gut und Böse

Die einfachste und grundlegendste Beobachtung aber ist die: Sony und Microsoft haben mit ihren Investitionen in die aktuelle Konsolengeneration Milliarden verloren, die sie nie wieder hereinholen werden. Sie können diese Verluste gegenüber ihren Aktionären nicht vertreten, sie können sie nicht wettmachen – warum also sollten sie jemals eine weitere Konsole entwickeln? Wenn Sony sagt, dass die Lebensdauer der Playstation 3 auf zehn Jahre angesetzt ist, dann meinen sie in Wirklichkeit, dass sie es schlicht nicht vertreten können, in absehbarer Zeit eine weitere Konsole zu produzieren. Wenn man Microsoft fragt, welcher Verantwortliche im Haus die nächste Generation der Xbox entwirft, erhält man keine Antwort.

Und Nintendo?

Wenn ich all diese Punkte anführe, fragen die Leute zwangsläufig: »Okay, aber was ist mit Nintendo?« Nintendo ist die Ausnahme, die die Regel bestätigt. Nintendos Wii ist die erfolgreichste und einzig profitable Konsole der aktuellen Generation im Markt. Warum? Weil Nintendo aus veralteten Hardware-Bausteinen ein

World of Warcraft hat den Spielemarkt auf dem PC umgekrempelt. Sein effizientes Gebührenmodell schützt es effektiv vor Raubkopien.

Die asiatischen Märkte werden längst von Online-Spielen wie **Maple Story** dominiert.

Die Ausnahme von der Regel: **Nintendos Wii** ist günstig und rentabel, weil sie auf innovative Steuerung statt High-End-Grafik setzt.

Der Konsolenmarkt in den USA

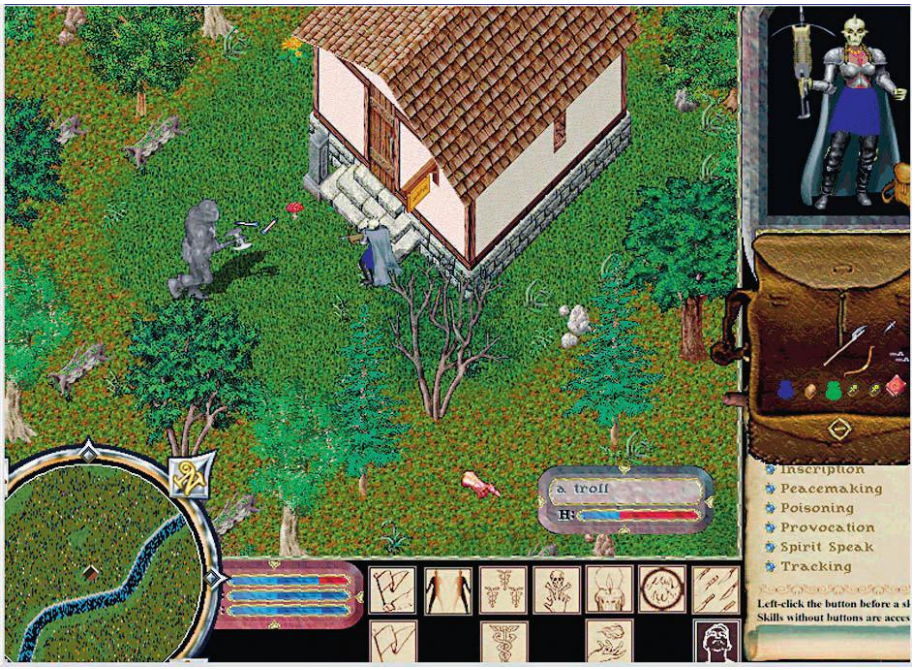
Hardware: Verkaufte Einheiten 2007

Nintendo Wii	XBox 360	Playstation 3
6,29 Millionen	4,62 Millionen	2,56 Millionen

(Quelle: NPD Group)

Spiele: Verkaufte Einheiten (Dezember 2007)

Platz	Spiel	System	Stückzahl
1	Call of Duty 4	XBox 360	1,47 Millionen
2	Super Mario Galaxy	Wii	1,40 Millionen
3	Guitar Hero 3	Playstation 2	1,25 Millionen
4	Wii Play	Wii	1,08 Millionen
5	Assassin's Creed	XBox 360	893.700
6	Halo 3	XBox 360	742.700
7	Brain Age 2	DS	659.500
8	Madden NFL 08	Playstation 2	655.200
9	Guitar Hero 3	XBox 360	624.600
10	Mario & Sonic at the Olympic Games	Wii	613.000



Ultima Online war das erste Online-Rollenspiel – und zugleich das erste Spiel auf Basis von DirectX.

günstiges System zusammengestellt und stattdessen in ein innovatives neues Steuerungsgerät investiert hat. Was ist bedeutsam an dieser Entscheidung? Nintendo hat diesen Überlebenstrick aus der Zeit gelernt, als ihnen schon einmal ein Markt weggestorben ist, der der Arcade-Automaten. Wenn man sich heute in einer Spielhalle umsieht, entdeckt man darin die kläglichen Überreste dessen, was in den 80er-Jahren mal das bestimmende Segment eines Multimilliarden-Dollar-Markts war, bevor gute Grafik erschwinglich wurde und das Alleinstellungsmerkmal einer 2.000 Dollar teuren Arcade-Maschine hinwegwischte. Was vom Automatengeschäft noch übrig ist, sind Spiele, die auf teuren, proprietären Steuerungsgeräten wie Lichtpistolen, Motorrädern, Skiern, Tanzflächen etc. basieren. All diese spezialisierten Geräte sind noch zu teuer, um sie an Haushalte zu verkaufen.

Woran erkennen wir also, dass die Konsolen sterben? Daran, dass die einzig profitable, dominante Konsole aus billigen Standardkomponenten besteht, und dass fünf der zehn bestverkauften Konsolenspiele im Weihnachtsgeschäft des letzten Jahres besondere Eingabegeräte voraussetzen (siehe Tabelle). Einer der großen Verkaufshits 2008 ist übrigens Rock Band, ein Spiel, für dessen Steuerungs-Hardware die Käufer 150 Euro hinblättern müssen.

Am PC führt kein Weg vorbei

Vor vielen Jahren habe ich bei Microsoft eine Strategie ins Rollen gebracht, um den PC und mit ihm Windows in die Haushalte und letztendlich ins Wohnzimmer zu bringen. Der ursprüngliche Codename für das Projekt, aus dem schließlich DirectX wurde, lautete »Manhattan Project«. In meiner jugendlichen Begeisterung wog die political incorrectness dieser Bezeichnung damals wenig gegenüber der Ironie, die in ihr steckte, denn immerhin sollte das supergeheime Untergrundprojekt sowohl den Medienkrieg zwischen dem PC und Apple ein für alle mal beenden, als auch den PC in die Position katapultieren, mit den erfolgreichen japanischen Konsolen von Sony, Nintendo und Sega konkurrieren zu können. Zu jener Zeit schrieb ich ein Strategiepapier namens »Taking Fun Seriously« für Microsoft, in dem ich argumentierte, dass der erfolgreichste Weg hin zur Dominanz von Windows in Heim und Wohnzimmer darüber führe, Spiele-Unterstützung zum zentralen Element des Betriebssystems zu machen. Der entscheidende Punkt dieses Strategiepapiers von 1994 war, dass Windows soziales Spielen ermöglichen und damit den PC zur dominanten Spieleplattform machen kann.

»Die Verbindung mit anderen Spielern wird eine der mächtigsten und mitreißendsten Revolutionen für Spiele sein. Die meisten Spiele sind isolationistisch. Bestimmte Arten von Fantasie benötigen Privatsphäre, aber die meisten Formen von Realitätsflucht sind sozial. Wettbewerb und Entdeckungsfreude machen mehr Spaß, wenn reale Menschen daran beteiligt sind. Die bestverkauften Konsolenspiele sind Kampfduelle zwischen zwei oder mehr Spielern. Das bestverkaufte PC-Spiel ist Doom, das im Netzwerk am meisten Spaß macht. Gemeinsam mit anderen Technologien könnte die Konnektivität das Geschäftsmodell der Spieleindustrie grundlegend umwälzen, fort von 50 Dollar teuren Wegwerf-Applikationen hin zur Entwicklung riesiger, erweiterbarer Server-Universen für zahlreiche Spieler. Multiplayer-Spiele könnten auf Set-Top-Boxen ein größeres Geschäft werden als Video-on-Demand.

Soziale Interaktion findet außerdem breiteren Zuspruch. Erforschbare Mehrspieler-Universen sind wahrscheinlich attraktiver für Frauen als herkömmliche, konfliktbasierte Solo-Titel.«

(Quelle: Microsofts internes Strategiedokument von 1994 zur Entwicklung von DirectX. Eine Randbemerkung: Ultima Online, das erste Online-Rollenspiel mit Massenmarkt-Erfolg, war gleichzeitig die erste echte DirectX-Anwendung. Die zweite war Blizzards Diablo.)

Zehn Jahre, nachdem DirectX geschaffen worden war, um einen Markt für Online-Spiele zu ermöglichen, wurde World of Warcraft zum profitabelsten Spiel aller Zeiten, mit einem Umsatz von mehr als 1,2 Milliarden Dollar pro Jahr – das ist mehr als der Umsatz aller Xbox-Spiele zusammen. World of Warcraft erscheint ausschließlich für den PC. Der PC ist heute die größte Spieleplattform der Welt, auf der mehr Menschen spielen und auf der mehr Umsatz erwirtschaftet wird als auf allen Konsolen zusammen. PC-Spiele in Schachteln verschwinden auch deshalb, weil World of Warcraft mit einem Schlag Dutzende von Spielen überflüssig gemacht hat; es unterhält über einen Zeitraum, für den man normalerweise zahlreiche andere Spiele hätte kaufen müssen. World of Warcraft und andere Online-Spiele gibt es inzwischen direkt als Download. Warum sollte man überhaupt noch ein Spiel im Laden kaufen?

Die Zukunft des Spielemarkts

Nach einer Studie der PC Games Association (PCGA) übertraf der weltweite Umsatz mit PC-Software und Hardware im Jahr 2007 57 Milliarden Dollar. Das Erstaunliche angesichts dieser überwältigenden Dominanz des PCs auf dem Spielemarkt ist die weit verbreitete Ansicht, dass es sich um eine sterbende Plattform handle, und dass alles Wachstum und alle Zukunft im Konsolengeschäft liege. In Wahrheit erlaubt es die offene Struktur der Plattform PC dem Spielemarkt, das antiquierte Einzelhandelsgeschäft mit seinen Spieleschachteln durch ein diversifizierteres Online-Modell zu ersetzen, das Ertrag aus einer Kombination aus Direktverkauf, Werbung, Gebühren und Mikro-Transaktionen erwirtschaftet. Viele Menschen halten das Verschwinden von PC-Spielen aus den Händlerregalen für ein Versagen am Markt, tatsächlich aber haben viele PC-Spiele den Bedarf für Schachteln und Kunststoffscheiben hinter sich gelassen und werden mittlerweile als komplett elektronisches Produkt vertrieben. Online-Geschäftsmodelle wie die moderner PC-MMOGs haben sich als extrem effizient erwiesen, wenn es darum geht, aus ihrer Spielerschaft Umsatz und Spielzeit zu saugen. Sie zerstören damit die Nachfrage nach Solo-Spielen. Der Erfolg in Asien und anderen Wirtschaftszonen lässt darauf schließen, dass MMOG-Modelle über kurz oder lang alle wesentlichen Spielemärkte dominieren dürften. Die einzigen Faktoren, die das Wachstum dieses Zweigs in westlichen Ländern bisher limitiert haben, sind vielerorts noch zu geringe Bandbreiten bei Internet-Zugängen und das Fehlen sicherer, allgemeingültiger Online-Zahlungslösungen. Beides ändert sich rapide – und zwar rapide genug, um den Bedarf für eine neue Konsolengeneration auszulöschen, noch bevor die jetzige sich überlebt hat.

Alex St. John

Guitar Hero ist auch deshalb so erfolgreich, weil es auf ein proprietäres Steuergerät setzt.

»Man kann mit 43 MB viel anfangen«

Im Gespräch mit David Braben

Der Elite-Erfinder und Frontier-Chef gibt Einblicke in den Entstehungsprozess seines ebenso innovativen wie erfolgreichen WiiWare-Launch-Titels Lost Winds und äußert sich zum Hype um Online-Vertriebsplattformen.

David Braben

ist Firmengründer und Geschäftsführer von Frontier.

David Braben schuf mit Elite einen Meilenstein der Spielegeschichte. 1993 gründete er das Entwicklerstudio Frontier, bei dem er bis heute Geschäftsführer ist. Zu seinem Team gehören inzwischen über 180 Mitarbeiter, der bisher größte kommerzielle Erfolg ist Rollercoaster Tycoon 3. Derzeit arbeitet er am Action-Thriller The Outsider.

> Als Nintendo am Ende des letzten Jahres in einem reichlich mysteriösen Meeting mit uns über WiiWare sprach, hatten wir die Idee zu Lost Winds bereits in der Tasche.

> Die Arbeit an der Steuerung begann schon sehr früh, lange bevor wir ein richtiges Spiel hatten.

Making Games Hallo David. Schön, dass du für uns Zeit hast. Wie seid ihr zu der Ehre gekommen, mit Lost Winds einen Launch-Titel für WiiWare zu entwickeln?
David Braben Ehrlich gesagt: Das war zum Großteil einfach nur Glück. Als Nintendo nämlich am Ende des letzten Jahres in einem reichlich mysteriösen Meeting mit uns über WiiWare sprach, hatten wir die Idee zu Lost Winds bereits in der Tasche.Wir haben bei uns in der Firma etwas, das wir den Spiel-der-Woche-Prozess nennen – ein internes, geschlossenes Forum, in dem wir neue Spielideen entwickeln, auseinander nehmen und verbessern. Dabei sind wir ziemlich kritisch. Eine Idee, die es durch dieses Forum schafft, kann also nicht allzu schlecht sein. Und so ist auch die Idee zu Lost Winds entstanden.

Making Games Es gab also für Lost Winds keinen klassischen Pitch bei Nintendo?
David Braben Nein, darum geht es bei WiiWare aber auch gar nicht. Es gibt keinen Pitch, es gibt aber auch keine finanzielle Unterstützung. Wir verdienen erst am Verkauf, aber das lohnt sich. Mehr darf ich über das WiiWare-Geschäftsmodell von Nintendo aber nicht verraten.

Making Games Was macht denn WiiWare aus deiner Sicht zu einer attraktiveren Plattform als Xbox Live Arcade oder das Playstation Network?
David Braben Mit der Frage haben wir uns bei Lost Winds nie auseinander gesetzt. Das Spiel war schon für die Wii gedacht, seit wir zum ersten Mal von der Steuerungsmethode der Konsole gehört hatten, also quasi seit wir die ersten Nintendo Revolution Entwickler-Kits bekamen. Damals sollte die Wii ja noch Revolution heißen. Das dann auch noch WiiWare kam, war ein glücklicher Zufall, denn eigentlich sollte das Spiel ganz traditionell in die Läden kommen.

Making Games Die Wii hat euch also erst auf die eigentliche Idee zum Spiel gebracht?
David Braben Es war so: Unser Designer Steve Burkis hatte die Idee, eine Figur durch die Gegend zu blasen. Und dann ist diese Idee gewachsen. Wir fragten uns etwa, wie es wohl wäre, wenn ein Spieler die Figur und der andere den Wind steuert. Die logische Schlussfolgerung all dieser Vorschläge war, das Spiel für die Wii und den Motion Controller zu entwickeln.
Making Games Was muss ein deutsches Entwicklerstudio generell beachten, wenn es für WiiWare entwickeln will?
David Braben Zunächst einmal muss man als Wii-Entwickler bei Nintendo registriert sein. Aber dann unterscheidet sich der Entwicklungsprozess eigentlich nicht von dem anderer Spiele. Du versuchst einfach, deinen Spielern die verschiedenen Elemente des Spiels zu erklären. Die Hürden sind die Gleichen wie bei einem Spiel aus der Box. Für WiiWare zu entwickeln ist allerdings generell sehr gradlinig.

Making Games Wie viele Leute haben bei Frontier an Lost Winds gearbeitet?
David Braben Das Kernteam bestand aus zwölf Personen. Zum Ende hin haben wir natürlich auch ein wenig auf Hilfe von außen zurückgegriffen und Freelancer engagiert.

Making Games Kannst du uns auch verraten, wie lang der Entwicklungsprozess war?
David Braben Wir haben Weihnachten 2007 angefangen. Es war also eine recht kurze Entwicklungszeit von nicht mal sechs Monaten. Und das, obwohl wir in dem Zeitraum auch erst den Prototypen für die Steuerung entworfen haben.

Making Games Danke fürs Stichwort: Ihr greift für Lost Wind stark auf die Bewegungserfassung der Wii zurück. Wie lange hat die Perfektionierung der Steuerung gedauert?
David Braben Die Arbeit an der Steuerung begann schon sehr früh, lange bevor wir ein richtiges Spiel hatten. Wir haben dementsprechend mit ganz einfachen Platzhalter-Grafiken gearbeitet, nur um die Steuerung letztendlich so hinzubekommen, dass sie sich richtig anfühlt. Schließlich ist die Steuerung das Wichtigste am ganzen Spiel, vor allem bei einem Wii-Projekt.

Making Games Wann habt ihr euer Spiel zum ersten Mal Nintendo gezeigt?

David Braben Wir haben viel über das Spiel mit den Leuten von Nintendo gesprochen, aber wirklich gesehen haben sie es tatsächlich erst kurz, bevor es fertig war.

Making Games Euch standen für Lost Winds nur 43 Megabyte Speicherplatz zur Verfügung, trotzdem habt ihr ein grafisch verhältnismäßig aufwändiges Spiel entwickelt, das unter anderem Reflektionen sowie Bewegungs- und Tiefenunschärfe beherrscht. Wie habt ihr so viele Daten in so wenige Megabyte gepresst?
David Braben Das ist immer eine Frage der Perspektive. Bei Elite musste ich damals mit 22 Kilobytes arbeiten, die meisten E-Mails sind heute wegen des ganzen Formatierungsmülls größer als das. Wenn man mir damals gesagt hätte, dass mein Spiel 43 Megabyte groß sein darf, hätte ich vor Freude aufgeschrien. Man kann verdammt viel anstellen mit 43 Megabyte. Ich glaube, Lost Winds ist sogar kleiner als das. Ich will hier nicht die Speicherbegrenzungen bei WiiWare kommentieren. Die Wahrheit ist doch, dass wir immer mit irgendwelchen Limits arbeiten. Wenn es mal nicht die Festplatte ist, dann ist es eben der Arbeitsspeicher oder die Zahl der Polygone, die wir nutzen können. Es geht immer ums Optimieren, ums Tunen, und das war natürlich auch bei Lost Winds wichtig.

Making Games Deine Jungs mussten also einfach nur lernen, schlank und effizient zu programmieren? Das ist das Geheimnis?
David Braben Ganz genau. Stell dir doch nur mal vor, wie viele Bücher in 43 Megabyte passen. Das wäre ein ausgesprochen langes Bücherregal. Also ja: Solche Arbeitsbedingungen sind ein guter Lehrmeister, denn wir müssen wie gesagt immer mit Limits arbeiten.

Making Games Einige Spiele erscheinen sowohl auf dem Playstation Network als auch für Xbox Live Arcade. Glaubst du, dass diese systemübergreifende Entwicklung ein Trend bei Online-Vertriebsplattformen wird?
David Braben Das ist wie mit allen Entwicklungen immer eine Frage der Angemessenheit. Aber es würde mich überraschen, wenn sich dieser Trend nicht fortsetzen würde. WiiWare, Xbox Live Arcade und Playstation Network sind doch im Grunde nichts anderes als ein alternativer Vertriebsweg für Konsolenspiele. Dass die Spiele bisher noch relativ klein sind, liegt nur an den begrenzten Übertragungsgeschwindigkeiten. Aber das wird sich sicherlich bald ändern.

Making Games Sind Online-Vertriebsplattformen für kleine, unabhängige Entwickler besonders attraktiv, weil ihre Spiele so leichter auf den Markt kommen?
David Braben Sicher, diese Systeme sind eine große Chance für kleinere Entwicklerteams. Ich freue mich zum Beispiel sehr auf World of Goo. Es ist toll, dass man ein Spiel so leicht veröffentlichen kann. Aber es gab immer Möglichkeiten, kreativ zu sein. Deshalb ist es für die Entwickler letztendlich wichtiger, eine Vorstellung davon zu haben, wohin sich Spiele in der Zukunft entwickeln werden, statt sich Gedanken über den Vertriebsweg zu machen.

Nur allein wegen der Download-Plattformen werden Spiele weder innovativer noch experimenteller. Ein Spiel wie Lost Winds hätte es aber auf dem klassischen Vertriebsweg sehr schwer gehabt, weil sich die Spielmechanik noch nie beweisen konnte. Aber über die Online-Verkäufe konnten wir sehr gut sehen, wie viele Leute diese Art von Spiel überhaupt mögen. Früher hätte ein Publisher ein hohes finanzielles Risiko mit uns eingehen müssen. Jetzt liegt das Risiko bei uns, aber wir mussten nicht so viel investieren, weil die Entwicklungszeit recht kurz war. Lost Winds ist zum Glück ein sehr fokussiertes, gradliniges Spiel und deshalb auch kalkulierbar.

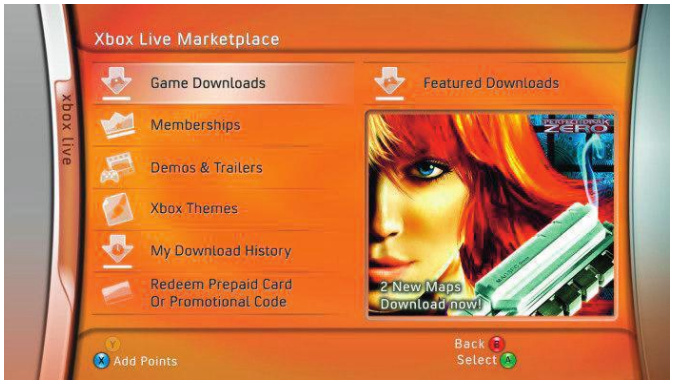
Making Games Sollten Online-Vertriebssysteme eine Art finanzielles Rückrat für unabhängige Entwicklerstudios sein?
David Braben Warum nur für unabhängige Entwickler? Die Online-Plattformen sind eine

»Bei WiiWare gibt keinen Pitch, es gibt aber auch keine finanzielle Unterstützung. Wir verdienen erst am Verkauf, aber das lohnt sich.«

»Bei Elite musste ich damals mit 22 Kilobytes arbeiten, die meisten E-Mails sind heute wegen des ganzen Formatierungsmülls größer als das.«

»Lost Winds war schon für die Wii gedacht, seit wir zum ersten Mal von der Steuerungsmethode der Konsole gehört hatten, also quasi seit wir die ersten Nintendo Revolution Entwickler-Kits bekamen.«

» Dass die Spiele bisher noch relativ klein sind, liegt nur an den begrenzten Übertragungsgeschwindigkeiten. Aber das wird sich sicherlich bald ändern. «



»Ich glaube beispielsweise, dass Xbox Live Arcade zu viele neue, aber mittelmäßige Spiele anbietet. Auch die meisten Remakes sind nicht sonderlich toll.«

Chance für jeden. Man kann neue Ideen ausprobieren, und die Spieler bekommen diese neuen Ideen zu sehen. Die Entwicklung von neuen innovativen Spielelementen stagniert doch im Moment. Es gibt nur noch Fortsetzungen, aber keine Originale mehr.

» Früher hätte ein Publisher ein hohes finanzielles Risiko mit uns eingehen müssen. «

Making Games Verändern die Online-Plattformen die grundsätzlichen Arbeitsweisen von unabhängigen Entwicklerstudios?
David Braben Ich will die Systeme nicht schlecht machen. Es ist toll, dass wir auf die unterschiedlichen Plattformen zurückgreifen können, aber die Entwicklung eines WiiWare-Spiels unterscheidet sich nicht sonderlich von der eines normalen Wii-Titels. Aber es gibt einen klaren Vorteil auf der kreativen Seite. Dank des schlanken Geschäftsteils können wir mehr Risiken eingehen. Wir können Ideen ausprobieren, die die Spieler uns im Laden vielleicht nicht abnehmen, online aber ausprobieren.

» Wir wollen keinen Müll entwickeln, wir wollen stolz auf unsere Arbeit sein. «

Making Games Stecken im Hype um die Online-Plattformen auch Gefahren?
David Braben Auf jeden Fall! Ich glaube beispielsweise, dass Xbox Live Arcade zu viele neue, aber mittelmäßige Spiele anbietet. Auch die meisten Remakes sind nicht sonderlich toll.

»Ich freue mich zum Beispiel sehr auf World of Goo. Es ist toll, dass man ein Spiel so leicht veröffentlichen kann.«

Es ist einfach schwer, gute Titel zu finden. Ich kann zwar Demos runterladen, und dass ist generell toll, aber ich will mein Leben auch nicht mit miesen Demos vergeuden. Ich würde lieber durch ein Filtersystem wissen, was andere Spieler von den Titeln halten.

Was mir bei WiiWare gleich gefallen hat, war der Nintendo Channel und die Möglichkeit Spiele zu bewerten. Das ist fast wie bei Youtube, und ich halte es für einen großen Vorteil. Es geht nicht um große oder kleine Publisher. Es geht darum, Müll so deutlich zu machen, dass die schlechten Spiele in der Versenkung landen und die Benutzer sich den Schrott nicht mehr runterladen und kaufen müssen.

Publisher, Produzenten und alle, die an der Entwicklung eines Spiels beteiligt sind, sollten lernen, dass Spiele gut sein müssen, um sich zu verkaufen. Im Moment gibt es einfach zu viele schlechte Spiele, die sich nur gut verkaufen, weil sie eine tolle Verpackung haben oder weil die liebsten Zeichentrickfiguren der Kinder im Namen auftauchen. Eltern kaufen diese schlechten Spiele und die Kinder sagen ihnen dann, wie blöd das Spiel ist. Das Problem ist, dass die Eltern daraus nicht lernen, gute Spiele zu kaufen, sie kaufen einfach überhaupt keine mehr. Bewertungssysteme wie beim Nintendo Channel nützen auch den Entwicklern. Wenn wir uns der Qualität unserer Spiele bewusst sind, stärkt das unser Selbstbewusstsein und wir steigen auf. Der Müll fällt hoffentlich zu Boden.

Making Games Deine Hoffnung auf mehr Innovationen in Spielen beruht also auf den Nutzerbewertungen von Spielen?
David Braben Ja, denn so steigt der Druck, wirklich gute Spiele zu entwickeln. Und das ist doch letztendlich das, was wir alle wollen. Wir wollen keinen Müll entwickeln, wir wollen stolz auf unsere Arbeit sein. Nintendo macht das bereits mit der eigenen Marke. Spiele von Nintendo sind in der Regel sehr gut. Bei anderen Firmen und Marken weiß ich dagegen nie, ob ich ein gutes Spiel bekomme oder nicht.

Making Games Gibt es goldene Regeln für die Spieleentwicklung für Online-Plattformen?
David Braben Nein, die gibt es nicht. Goldene Regeln sind überhaupt sehr gefährlich, weil sie dich leicht einschränken. Es gibt die bizarrsten Spielideen, wie zum Beispiel Portal. Diese Ideen wirken im ersten Moment unmöglich, aber sie eignen sich vielleicht perfekt für die Online-Vermarktung. Einige Leute werden denken, das Spiel ist merkwürdig. Anderen wiederum wird es gleich fantastisch gefallen. Es gibt also nur eine Regel und die ist eigentlich keine Regel, sondern etwas Offensichtliches: Es muss gut sein.

Heiko Klinge

Jobs @ IDG

IDG – das IT-Medienhaus

IDG ist das weltweit führende IT-Medien-Haus mit Präsenz in rund 80 Ländern und einem Umsatz von über 2,5 Mrd. US-Dollar. Mit rund 80 Millionen Pageimpressions pro Monat gehören die Internetangebote der IDG Communications Media AG zu den reichweitenstärksten Special-Interest- und Fachinformationsangeboten im deutschsprachigen Internet.

Seit 40 Jahren steht unser Name für Top-Qualität in der Technologie-Information. Und das in ihrer gesamten Breite. Tochtergesellschaften der deutschen IDG Communications Media AG sind die IDG Business Media GmbH, IDG Magazine Media GmbH und die IDG Entertainment Media GmbH. Mit unseren Titeln, Events, Kongressen und Mediendienstleistungen erreichen wir alle IT-Zielgruppen – vom Computer-Anwender über PC- und Konsolen-Spieler bis zum IT-Entscheider.

IDG – das vernetzte Medienhaus

Wie erreichen Informationen die Zielgruppe? Mit seinen Online-Formaten und Print-Publikationen sorgt IDG für den wichtigen Wissenstransfer. Die Events und Seminarreihen tragen dazu bei, die Zielgruppen persönlich untereinander zu vernetzen. Darin sind wir Profis. Denn wir vernetzen uns auch intern…

… IDG Parkstadtgespräche geben einen „Blick über den Tellerrand“. Es kommen regelmäßig außergewöhnliche Persönlichkeiten zu Wort. Ob Extrembergsteiger, Hirnforscher oder„Zeit-Manager“ – stets bieten die Vorträge und Diskussionen Einblick in Themenfelder abseits des beruflichen Tagesgeschäftes.

… IDG Happy Hour als Plattform zum Austausch mit Kollegen und Kolleginnen, die einem im Tagesgeschäft nicht so oft begegnen.

… IDG Projektgruppen, die uns schon oft neue Ideen, Produkte und Impulse gegeben haben und so unser Wachstum fördern.

Talents @ IDG

Wenn Sie sich trauen, Dinge zu bewegen, sich einbringen und vernetzen und auf alle Fälle eine Prise Humor besitzen, dann sind Sie genau die/der Richtige für uns!

Klicken Sie sich zu uns unter www.idgtalents.de um mehr über unsere Philosophie zu lesen. Klicken Sie sich auf www.idgmedia.de um unsere aktuellen Stellenangebote zu lesen. Denn wir suchen in allen Bereichen Persönlichkeiten als

- **Key Account Manager (m/w)**
- **Netzwerkadministrator mit Schwerpunkt Web Systeme (m/w)**
- **Web Developer für die Redaktion CIO (m/w)**
- **Assistent/innen**

Weiterhin suchen wir aktuell Sales-Trainees und ermöglichen mit dem dualen Nachwuchs-Entwicklungsprogramm “Work & Study” in Kooperation mit der BAW München innerhalb eines Jahres einen Abschluss zum Sales-Fachwirt sowie auf Wunsch gerne die Bachelor Thesis.

making games TALENTS

IDG Communications Media AG - Human Resources
Lyonel-Feininger-Str. 26 – 80807 München – Telefon 089/ 360 86 – 240
info@idgtalents.de – www.idgmedia.de – www.idgtalents.de

Der Ferrari auf dem Rübenacker
Effiziente Projektstrukturen

Nicht weiterblättern! Dies ist nicht schon wieder ein Artikel über Projektmanagement. Ja, das P-Wort kommt vor. Aber dieser Artikel handelt von den Voraussetzungen und Strukturen, damit es überhaupt funktionieren kann.

Erik Simon

ist Management und Production Consultant.

Erik Simon arbeitet seit 1988 in der Spielebranche. Er war Entwicklungsleiter bei Blue Byte, JoWooD und zuletzt bei Sunflowers, wo er die Entstehung von Anno 1701 begleitete. Weitere bekannte Titel, an denen er leitend mitgearbeitet hat: Die Siedler 3 und 4, Gothic 2 sowie Spellforce. Erik Simon sammelte sowohl Erfahrungen als Grafiker wie auch als Designer und Projektleiter, bis er sich schließlich auf das Entwicklungs-Management spezialisierte.

Wahrscheinlich haben sich die meisten Entwickler schon einmal mit dem leidigen Thema Projektmanagement auseinandergesetzt. Es hat sich rumgesprochen, dass gutes Projektmanagement den kreativen Erfolg in den heutigen komplexen, termingebundenen Projekten überhaupt erst möglich macht. Und so bewaffnen sich die Studios mit Standardwerken des Projektmanagements, stellen Projektmanager ein und werfen bald mit Ausdrücken wie Wasserfall-Modell, Staged Delivery und Agile Development um sich.

Und dann scheitert wieder ein Projekt oder verspätet sich oder ist weit über Budget. Und dem Scherbenhaufen, der dann vor einem liegt, sieht man oft nicht so recht an, warum er in die Brüche gegangen ist: War das Team schuld? Die Umstände? Der Publisher? Der Projektmanager? Oder die Projektmanagement-Methode? Und weil man das oft nicht genau herausbekommt, ist es schwer, daraus zu lernen. Deshalb ist es wichtig, dem Projektmanagement den richtigen Boden zu verschaffen. Nur so findet man den Ansatz, der in einer bestimmten Konstellation von Team und Projekt funktioniert. Denn es erscheint zunächst verdammt schwer, im weiten Feld von Projektmanagement-Methoden diejenige zu finden, die unter den gegebenen Umständen die beste ist. Nach vielen Jahren der Beschäftigung mit dem Thema wage ich zu behaupten: Fast alle Methoden funktionieren – solange man sie nur konsequent genug einsetzt!

Und deshalb ist es wichtig, dem gewählten Projektmanagement-Ansatz den richtigen Boden zu bereiten. Denn ansonsten nützen die besten Projektmanagement-Methoden nichts; sie kommen nicht voran und geben ein schlechtes Bild ab – so wie ein Ferrari auf dem Rübenacker.

Die zwei Übeltäter

Es gibt zwei große Themenbereiche, die auch das beste Projektmanagement zum Scheitern bringen können: Entweder die leitenden Mitarbeiter im Team sind nicht vom Prinzip überzeugt, Kontrolle auszuüben. Oder die äußeren Umstände und die zeitliche Struktur des Projektes stimmen nicht und lassen Projektmanagement zu einem Kampf gegen Windmühlenflügel werden. Auf beide Bereiche werden wir einen praxisorientierten Blick werfen, auf den zweiten besonders ausführlich. Zugegeben: In gewisser Weise ist dies doch ein Artikel über Projektmanagement. Aber wer schon einmal so weit gelesen hat, schafft den Rest auch noch. Und wird die investierte Zeit vielleicht um ein Vielfaches zurückbekommen.

Der innere Widerwille

Über den ersten Themenbereich, den inneren Widerwillen, der Projektmanagement zum Scheitern bringen kann, wird selten gesprochen. Man geht davon aus, dass die leitenden Mitarbeiter eines Studios mit der Zeit gehen. Dass ihnen klar ist, dass sie der wachsenden Komplexität moderner Spieleprojekte nur mit einem gesunden Maß an Organisation und ein paar guten Prozessen Herr werden können. Was aber, wenn dem nicht so ist? Die Ursachen für eine solche innere Verweigerungshaltung sind vielfältig: Manche fähigen kreativen Köpfe sind im Laufe der Zeit in leitende Positionen gerutscht, deren Jobinhalte nicht zu ihren Talenten passen. Old-School-Entwickler schaffen manchmal nicht den Schritt weg von lieb gewonnenen, aber chaotischen Entwicklungsmethoden. Oder es wird die Energie und Disziplin, die für Projektmanagement-Prozesse benötigt wird, nicht aufgebracht, weil man schlicht und ergreifend keine Lust dazu hat.

Projektstruktur: Die wichtigste Basis

PPP	Preprod	Main Production	Alpha	Beta
	Prototyp(en)			
Grundidee	Detailed Design			
Hauptrisiken und Preproduction definieren	Gemeinsame Detailspezifikation	Milestones		
Design / Planung Preproduction	Projektplan, Milestones	Vordefinierte Redesignphasen		
Schätzung Budget Prepod / Main Prod	Definition von Zeit- und Geld-Budget	Alpha: Feature Complete	Beta: nur Debugging	Gold!

Das Problem ist, dass man heute dem Thema Projektmanagement kaum ausweichen kann. Es wird vom Publisher und oft auch von der Studioleitung gefordert. Also werden die erforderlichen Prozesse ausgeführt, Tools benutzt und Dokumente geschrieben. Aber Projektmanagement funktioniert nur, wenn es zumindest vom Projektmanager, der Studioleitung und allen Leads ehrlich gelebt wird. Die einzelnen Prozesse sind nicht besonders schwer oder kompliziert, müssen aber regelmäßig und gründlich gemacht werden – gerade dann, wenn es stressig wird und man viel lieber konkret am Projekt basteln würde. Oft kommt es auch vor, dass leitende Mitarbeiter sich zu den Prozessen zwingen, ohne an deren Wirkung zu glauben und sich so selbst vormachen, Projektmanagement zu betreiben, während sie eigentlich nur die Form wahren wollen und die lästige Arbeit so schnell es geht hinter sich bringen wollen. Dieser innerlich eingestandene oder auch unbewusste Selbstbetrug ist eines der sichersten Mittel, Projektmanagement zu sabotieren. Und noch dazu kann man mit dem Finger auf den ungeliebten Prozess zeigen, wenn wieder etwas schief gegangen ist.

Das Hinterhältige an diesem Problem ist, dass es von außen oft schwer in kurzer Zeit zu erkennen ist. Hinreichend komplexe Projektpläne und andere Dokumente lassen sich auch mit dieser inneren Verweigerungshaltung erstellen. Aber da die Datenbasis der Dokumente oft Müll ist, werden die Ziele trotzdem verfehlt.

Innerhalb des Studios ist eine solche Einstellung einfacher zu erkennen und lässt sich zumindest dann bekämpfen, wenn die Studioleitung das ehrliche Ziel hat, die eigenen Entwicklungsprozesse verlässlicher zu machen. Aus der Publisherperspektive ist ein Fehlverhalten dieser Art schwerer wahrzunehmen – doch spätestens beim nächsten Milestone kommt unweigerlich die Stunde der Wahrheit.

Die Zeiteinschätzung

In einem guten Projektmanagement-Prozess kommen die Zeiteinschätzungen von den ausführenden Teammitgliedern. Diese wichtigste aller Datenbasen zu verbessern, gelingt nur, wenn man die Prozesse allen Teammitgliedern erklärt und darauf hinweist, dass Projektmanagement das Projekt kontrollieren will, aber nicht die Menschen. Man kann dem einzelnen Mitarbeiter übrigens sehr gut dabei helfen, seine Fähigkeiten in der Zeiteinschätzung zu verbessern, indem der Projektmanager ein Dokument über die Zeiteinschätzungen jedes Teammitgliedes führt und festhält, wie gut die Prognosen zutrafen. Diese Ergebnisse fließen an jeden Mitarbeiter zurück und erlauben ihm oder ihr einen objektiven Blick auf diese wichtige Fähigkeit.

Für die leitenden Mitarbeiter gibt es aber keine Anleitung, um Projektmanagement lieben zu lernen. Wenn der Leidensdruck verfehlter kreativer und finanzieller Ziele nicht ausreicht, um eine innere Abneigung gegen geplantes Vorgehen zu beseitigen, dann sollte man zumindest sich selbst gegenüber ehrlich genug sein und darüber nachdenken, ob man in kleineren Projekten nicht vielleicht zufriedener arbeiten kann, anstatt in großen komplexen Entwicklungen einen auf Dauer aussichtslosen Kampf zu führen. Denn eines ist gewiss: Die Zeiten, in denen erfolgreich einfach drauflos entwickelt werden konnte, sind vorbei.

Das Projekt statt des Produkts

Aber auch wenn studiointern keine Berührungsängste mit dem Thema Projektmanagement existieren, ist der Erfolg der Planungs- und Kontrollprozesse leider nicht gesichert. Gelungenes Projektmanagement hängt nämlich zu einem großen Teil von den geeigneten Rahmenbedingungen ab. Und hier liegt oft der Hase im Pfeffer. Der Finanzier und der Entwickler eines Projektes müssen sich über gewisse Grundgesetze der Projektentwicklung im Klaren sein und sich eindeutige Spielregeln geben, damit ein Projektmanagement überhaupt greifen kann. Und dieser erste, wichtige Schritt wird auch heute noch erschreckend oft unterlassen.

Es ist unmöglich, alle drei Faktoren Zeit, Budget und Qualität zu Beginn eines Projektes festzulegen. Man sollte meinen, dass sich dieses fundamentale, unumstößliche Gesetz des Projektmanagements mittlerweile überall herumgesprochen hat, bis in die Chefetagen der Publisher. Leider weit gefehlt. Kaum haben sich Entwickler und Finanzier weit genug angenähert, dass ein Projektstart möglich erscheint, wird auch schon gefragt: »Wann seid ihr fertig? Wie viel kostet das? Wo ist die Liste der Hauptfeatures?«

Wissen zu wollen, wie hoch die Kosten für ein Produkt sind, bevor man es kauft, erscheint zunächst wie ein verständlicher Reflex eines Geldgebers. In Wirklichkeit ist es aber ein sicheres

Design-Dokumente sollten schon während der Pre-Production so detailliert wie möglich ausgearbeitet sein.

Bei **Anno 1701** gelang es Related Designs und Sunflowers mit den hier vorgestellten Methoden, das Aufbau-Strategiespiel »in time«, »in budget« und »in quality« zu produzieren.

Auch wenn es hier von den Related-Designs-Mitarbeitern etwas übertrieben dargestellt wird: Regelmäßige Meetings und **Besprechungen** sind für effizientes Projektmanagement unumgänglich.

Mittel, um einen kontrollierten Ablauf unmöglich zu machen. Denn der Geldgeber kauft kein Produkt, sondern ein Projekt. Und ein Projekt ist gekennzeichnet durch die Einmaligkeit seiner Bedingungen. Wie also soll ein Entwickler all die Tausenden von Faktoren, die Kosten, Zeit und Qualität bestimmen, von vornherein wissen? Zumal sie oft noch durch Research herausgefunden werden müssen. Noch mal und bitte merken, lieber Publisher und lieber Studiochef: Es ist unmöglich.

Der Projektablauf

Um zu einem kontrollierten Projektablauf zu kommen, müssen sich Entwickler und Geldgeber auf eine Projektstruktur einigen. Diese sagt nicht nur aus, wann was fertig wird, sondern vor allem auch, wann welche Aussage über das Projekt gemacht werden kann. Diese Aussagen müssen dann aber wesentlich verlässlicher sein als das, was momentan in der Spielebranche üblich ist.

Werfen wir einen Blick auf eine solche Projektstruktur in der Praxis (Kasten auf Seite 54). Dort ist eine schematische Übersicht der Struktur abgebildet, auf die sich Sunflowers und Related Designs für die Produktion von Anno 1701 geeinigt haben.

❶ **PPP (Pre-Preproduction):** »Die Planung der Planung«. Hier entsteht ein Grund-Design, eventuell ein peppiges Publisher-Proposal und vielleicht eine erste grobe Schätzung des Umfangs, die aber als vorläufig und fehlerbehaftet deklariert wird. Vor allem aber wird der Prototyp geplant. Und das Wichtigste dabei ist, die Hauptrisiken des Projektes zu finden und einen Prototypen zu planen, der diese Risiken adressiert. So kann je nach Projekt der Inhalt des Prototypen komplett unterschiedlich ausfallen. Er kann etwa nur eine Technologie demonstrieren oder ein abstrakter Gameplay-Prototyp sein, wenn man eine ausgefallene Spielidee umsetzen will.

❷ **Preproduction:** Die vielleicht wichtigste Phase des Projektes, die sich trotzdem mit einem schlanken Team erfahrener Mitarbeiter erledigen lässt. Während der Prototyp erstellt wird, laufen gleichzeitig eine Menge umfangreicher Planungsarbeiten, die im Idealfall teilweise gemeinsam mit dem Publisher oder dem Geldgeber erledigt werden (dazu später mehr). Am Ende der Preproduction und mit Hilfe der Dinge, die man bei der Erstellung des Prototypen lernt, sollten folgende Planungsunterlagen fertig sein:

- Detailspezifikation: detailliert herunter gebrochene Beschreibung aller Features (z.B. mit Hilfe eines Mind-Manager-Tools)
- Detailed Design: komplette Ausspezifizierung all dieser Features
- kompletter Projektplan auf Basis dieser Dokumente
- Definition von Zeit- und Geldbudget
- Milestones

Es ist oft zu hören, dass sich in einem kreativen Produkt wie einem Spiel Features, Design und Projektplan unmöglich zu Beginn festlegen lassen. Diese Dokumente sind auch nicht dazu gedacht, unverändert zu bleiben. Oder gar dazu, einfach stur nach ihren Vorgaben drauflos zu produzieren. Mit Sicherheit werden sie alle im Laufe der Produktion große Veränderungen durchlaufen. Aber wenn man zu Beginn seine Hausaufgaben gründlich macht, können sie eines leisten: die Quantität der zu erledigenden Arbeit bestimmen. Die Risiken genau zu definieren. Und die Zeiträume zu umreißen, die man braucht, um eventuelle Unsicherheiten in den Griff zu bekommen.
Die je nach Projektgröße monatelange Arbeit, die im Prototypen und in den Planungen steckt, erlaubt einem erst, ein Budget zu errechnen. Dieses Budget sollte aber im Bewusstsein abgegeben werden, dass es sehr viel verlässlicher sein muss als allgemein üblich.

❸ **Main Production:** Das Team wächst und mit ihm die Anforderungen an das Management und das Projektmanagement. Aber die Leitlinien sind nun gesteckt und bereiten nicht noch zusätzlich Kopfschmerzen. Innerhalb des gewählten Entwicklungsmodells (Linear, Staged Delivery, Agile Development) kann man nun die Milestones anstreben, ohne den Gesamtüberblick zu verlieren. Vordefinierte Redesign-Puffer sorgen dafür, dass kritische Teile mehrfach implementiert werden können, falls nötig. Und auch für alle anderen Teile, die nicht so hinhauen wie ursprünglich gedacht, gibt der Projektplan vor, wie viel Zeit man für Arbeiten investieren kann, welche die ursprünglich geplanten Features ersetzen.

❹ **Alpha und Beta:** Hier sind im Kontext dieses Artikels eigentlich nur die Begriffsdefinitionen wichtig. Man sollte sich genau darauf einigen,

was Alpha und Beta bedeuten. In dem vorliegenden Fall heißt Alpha: Feature complete (Änderungen an vorhandenen, spielbaren Features noch möglich). Und Beta bedeutet: ausschließlich restliches Debugging.

Die Spielregeln

Eine Projektstruktur, auf die sich Entwickler und Geldgeber einigen, ist schon mal Gold wert. Aber es gibt auch noch weitere Faktoren, die eine kontrollierte Projektdurchführung unterstützen. Im Wesentlichen laufen diese darauf hinaus, sich im Vorfeld auf Spielregeln zu einigen und sich auch dann daran zu halten, wenn es stressig wird. Features gemeinsam spezifizieren: Wenn der Geldgeber über das geeignete Fachpersonal (Producer) verfügt, sitzen dieser Producer, der Game Designer und die Leads während der Preproduction regelmäßig an einem Tisch. Der Producer wird dabei als Sprachrohr des Publishers zu allen Features gefragt, soll sie jedoch nicht designen! Je detailreicher spezifiziert wird, desto reibungsloser läuft später die Produktion. Denn weil Entscheidungen im Vorfeld gemeinsam getroffen wurden, trägt man auch spätere Folgen von eventuellen Fehlentscheidungen gemeinsam. Man sollte darauf achten, die Features und Detailspezifikationen festzuhalten und abzeichnen zu lassen. Merke: Differenzen können während des Designs (billig) oder während der Produktion (teuer) beigelegt werden. Rollen und Verantwortlichkeiten müssen schriftlich definiert sein und nachgehalten werden: Genau wie es teaminterne Jobbeschreibungen und Definitionen von Verantwortlichkeiten geben sollte, gilt dies fast noch mehr für die Zusammenarbeit zwischen Entwickler und Publisher. Zwar meint jeder zu wissen, was die traditionellen Rollen, wie etwa die eines Producers, umfassen, aber in der Praxis werden wichtige Entscheidungen oft nicht getroffen, weil jeder Beteiligte meint, nicht dafür zuständig zu sein.

Hier hilft ein gemeinsam erdachtes Diagramm, in dem die Personen, ihre Aufgaben und Verantwortungsbereiche festgehalten sind. Das sieht zwar bürokratisch aus, hilft aber im täglichen Stress ungemein, wenn man darauf verweisen kann. Man kommt recht schnell zu einem solchen Diagramm, indem man die typischen, notwendigen Entscheidungsprozesse gemeinsam durchgeht und festlegt, wer in welcher Situation die Entscheidungen trifft. Beispiele dafür sind Milestone-Abnahmen und Eskalationswege im Falle von Problemen. So kann man verhindern, dass ein QA-Lead des Publishers zum Designer avanciert, ein Producer zum Marketing-Erfüllungsgehilfen wird und ein CEO durch Schweigen oder Feature-Änderungen alles durcheinander bringt.

Dem Projektmanager sollte es in dieser Struktur ausdrücklich erlaubt sein, sowohl beim Studio als auch beim Publisher sowie Leuten, die in der Hierarchie sowohl über als auch neben ihm stehen, auf Probleme und Fehlverhalten hinzuweisen. Denn alles, was mit Projektmanagement zu tun hat, kann nur nach dem Prinzip »Practice What You Preach« funktionieren.

Auch sehr nützlich ist es, Entscheidungen, die nicht in der täglichen Arbeit zwischen Team und Producer gelöst werden können, zu sammeln und der nächsten Hierarchiestufe (Entwicklungsleiter) an einem regelmäßig stattfindenden »Entscheidungstag« vorzulegen; wobei zuvor Alternativen erarbeitet wurden. Dies beschleunigt Entscheidungen erheblich, weil der Entscheider nur dann tiefer in die Materie eintauchen muss, wenn ihm keine der vorbereiteten Alternativen gefällt. Außerdem kann er den regelmäßigen »Entscheidungstag« in seinem Terminkalender reservieren und bremst Entscheidungen auf Grund von Abwesenheit seltener aus.

Der vorauseilende Gehorsam

Viele Arbeiten an einem Spiel werden von den notwendigen Marketing-, PR- oder Vertriebsarbeiten generiert. Nicht immer aber ist die Notwendigkeit des Vorausplanens von diesen Publisher-Mitarbeitern wirklich verinnerlicht worden, und so scheinen Messen und Presse-Events manchmal völlig überraschend über diese Abteilungen hereinzubrechen. Drei Wochen vor einer Messe kommen dann Anforderungen über 450 hochwertige Screenshots. Wenn es trotz aller Bemühungen nicht gelingt, solche Parteien in den Planungsprozess mit einzubinden, hilft nur, auf die eigene Erfahrung zurückzugreifen, Material im Vorfeld zu erstellen und die Zeiten dafür einzuplanen. Dieses Material stellt man mit zwei klaren Entscheidungsmöglichkeiten zur Verfügung, die bis in die Geschäftsleitung eskaliert werden: Soll das Spiel rechtzeitig fertig oder mehr Marketing-Material produziert werden?

Das Fazit

Mit dem hier vorgestellten Werkzeugkasten kann man die Vorarbeiten leisten, die dafür sorgen, dass das eigentliche Projektmanagement tatsächlich greifen kann. Es erfordert nur den Willen, sich von vornherein Regeln zu geben, anstatt darauf zu warten, dass spätere Konflikte diese Regeln im Nachhinein erzwingen. Und selbst wenn am Anfang das Verhältnis der beiden Partner noch gespannt sein sollte: Erfolgreiche Milestones und eine funktionierende Zusammenarbeit erzeugen schnell Vertrauen. Und das hilft dem Projektablauf ebenso wie ein gelungener Projektmanagement-Prozess. **Erik Simon**

Der **Producer** Christopher Schmitz (rechts) arbeitet für Ubisoft, plant aber vor Ort bei den Entwicklern (hier Related Designs) den Ablauf der Produktion.

Einsatzmöglichkeiten und Anwendungstipps
Hardware-Kopierschutz

Ungewollt ins Internet gelangte Software-Versionen verursachen schnell Schäden in Millionenhöhe. Ein hardware-basierter Kopierschutz inklusive Verschlüsselung per USB-Dongle kann das verhindern.

Steffen Kätsch

ist Leiter des technischen Supports bei MARX Software Security.

Steffen Kätsch absolvierte an der Fachhochschule Jena ein Elektronikstudium und wechselte dann zur MARX Software Security GmbH. Dort betreut der begeisterte Action- und Rennspiel-Fan die Kundenbetreuung sowie die Erstellung technischer Dokumentationen, arbeitet aber auch am technischen Design der MARX Hard- und Software mit.

Während das Internet aus dem Berufsalltag oder den heimischen vier Wänden gar nicht mehr wegzudenken ist, verfluchen Software-Firmen das digitale Medium immer häufiger. Verantwortlich dafür sind illegale Tauschbörsen, in denen regelmäßig geleakte Spieleversionen auftauchen.

Ob Preview-Versionen für Pressevertreter oder Alpha-Codes für den internen Gebrauch: Geraten solche Daten an die Öffentlichkeit, geht der Schaden oft in die Millionen. Allein im vergangenen Jahr beläuft sich die Schadenssumme in Deutschland laut BSA auf 1,2 Milliarden Euro. Weltweit verliert die Industrie laut einigen Schätzungen sogar bis zu 30,5 Milliarden Euro. Gerade im Zuge der Pre-Release-Phase entstehen vermehrt Sicherheitslöcher, die dem Unternehmen großen finanziellen Schaden zufügen können, sollte die Software schon vor der offiziellen Veröffentlichung über die einschlägigen Kanäle von Raubkopierern verteilt werden. Neben den eventuell geringeren Verkäufen drohen Entwicklern zudem saftige Schadensersatzansprüche der Vertragspartner.

Hierzu Dr. Andreas Lober, Rechtsanwalt und Partner bei der Frankfurter Anwaltskanzlei SchulteRiesenkampff: »Wenn ein Programm leakt, kann das durch geringere Umsätze für alle Beteiligten teuer werden. Falls der Entwickler daran schuld ist, haben dessen Vertragspartner – also in der Regel die Publisher – einen Schadensersatzanspruch, der in die Millionen gehen kann. Dann wird schnell relevant, ob man dem Entwickler Fahrlässigkeit oder gar grobe Fahrlässigkeit vorwerfen kann.«

Da einfache software-basierte Schutzmechanismen von versierten Crackern innerhalb kürzester Zeit ausgehebelt werden können, kommt für die Weitergabe von sensiblem Material nur eine Hardware-Lösung in Frage.

Digitales Sicherheitsschloss

Als gängigste hardware-basierte Kopierschutzlösung bieten Security-Firmen so genannte Dongles an, die von der Form her meist an USB-Sticks erinnern. Im Grunde verhindern Dongles aber nicht das Kopieren der Software, sondern beschränken vielmehr deren Funktionalität. Der Hersteller kann den Dongle beispielsweise so in seine Software integrieren, dass diese gar nicht erst startet, solange der Dongle nicht am Computer oder im Netzwerk angeschlossen ist. Ebenso ist es möglich, nur einzelne Funktionen wie Drucken oder Speichern zu unterbinden. Mit einem modernen Dongle können gleich mehrere Programme oder Programm-Module geschützt werden. Außerdem lassen sich Lizenzen im Netzwerk verwalten, was insbesondere die Kosten pro Lizenz für den Hersteller verringert. Dabei dürfen dann beliebig viele Lizenzvarianten verwendet werden: beispielsweise kann nur eine bestimmte Anzahl von Programmstarts erlaubt oder die Laufzeit der Software begrenzt werden.

Für den internen Schutz empfiehlt Korbinian Abenthum, Geschäftsführer des Entwicklers Weltenschmiede, den Einsatz eines Dongle-Systems, das nicht direkt mit dem Sourcecode zu tun hat: »Das wichtigste Feature ist die Automatisierung. Wir kompilieren die Anwendung

Moderne **Kopierschutz-Dongles** entschlüsseln Programmteile nur bei Bedarf und erschweren illegale Kopien.

jeden Tag mehrfach neu, und die muss sofort allen Scriptern zur Verfügung stehen.« Eine Kommandozeilenversion der Verschlüsselungs-Software des Dongles ist das A und O für die Spieleentwicklung. »Die automatische serverseitige Verschlüsselung der aktuellen Builds ermöglicht es, dass ein Scripter nur die neueste Version aus der Versionskontrolle holen muss und ohne jegliche Verzögerung weiterarbeiten kann.«

Auf technischer Ebene bieten aktuelle Dongles nicht nur einen auslesbaren Speicher, sondern ermöglichen hardware-basierte Ver- und Entschlüsselung innerhalb des im Dongle verbauten Controller-Chips. Bei Einsatz von leistungsfähigen und als sicher geltenden Verschlüsselungsalgorithmen wie AES ist es ohne den Dongle daher praktisch unmöglich, an die verschlüsselte Information zu gelangen. Dennoch: Die Vergangenheit hat gezeigt, dass sich jeder Schutz knacken lässt. Die Frage ist nur, ob sich der Aufwand für den Angreifer lohnt, die Software langwierig zu »cracken«. Mit einem modernen Dongle können nämlich zahlreiche Sicherheitsbarrieren eingebaut werden, die den Aufwand enorm steigern.

Dabei löst der Einsatz eines Dongles natürlich nicht automatisch alle Probleme, auch wenn das System sicher ist. Die Schwierigkeit für Security-Unternehmen besteht darin, dass meist nicht der Dongle selbst geknackt, sondern per Software das Verhalten der Hardware-Lösung emuliert wird. Dabei wird mitgeschnitten, was an Kommunikation zwischen Dongle und Programm stattfindet. Anschließend wird eine Bibliothek geschrieben, die sich zwischen die Kommunikation schiebt und die Abfrage anstatt des Dongles beantwortet. Darüber hinaus können findige Cracker auch die Dongle-Abfragen im Programm selbst patchen, um das Sicherheitssystem auszuhebeln.

Prävention ist alles

Um einem solchen Angriff schon von vornherein entgegenzuwirken, muss man sicherstellen, dass der Cracker den kompletten Hardware-Kopierschutz nicht auf einmal überblicken kann. Dazu verteilt man die Abfragen im gesamten Programm – auch an Stellen, an denen es der Cracker nicht erwartet. Die nächste Stufe stellen Dummy-Abfragen mit Crypt-Aufrufen sowie verschiedene Verschlüsselungen mit wechselnden Schlüsseln dar, die den Angreifer in die Irre führen. Das hilft vor allem bei Mitschnitten der Kommunikation zwischen Dongle und Programm. Besonders wichtig ist es, dass die Software nie komplett entschlüsselt im Speicher vorliegt, sondern immer nur Teile des Programms. Somit kann der Cracker die Daten nie auf einmal entschlüsseln und ihm ist letztendlich nur ein Teilerfolg beschert. Zum Beispiel kann jeder Level erst bei Bedarf entschlüsselt oder auch ein unterschiedlicher Schlüssel für unterschiedliche Level verwendet werden. Für das Patchen des Programms gibt es ebenfalls eine Lösung: Mit Hilfe von Anti-Debug-Techniken und Checksummenprüfungen von Programmteilen kann der Dongle bei Manipulationsversuchen gesperrt werden. Damit besteht für den Cracker latent die Gefahr, dass er sich bei Versuchen selbst aussperrt.

Hinsichtlich der Schutzmechanismen ist aber immer auch die Kreativität des Programmierers gefragt, der sein Programm kennt und weiß, was machbar ist. Je mehr Einfallsreichtum der Programmierer hier an den Tag legt, desto schwerer lässt sich die Software später cracken.

Vielfältiger Einsatzbereich

Ob für Preview-Versionen oder zum Schutz von Middleware wie KI-, Physik- oder Grafik-Engines: Ein Hardware-Kopierschutz bietet die größte Sicherheit. Dabei fallen die Kosten im Gegensatz zu einem möglichen Millionenschaden beim Durchsickern sensibler Daten verschwindend gering aus. Zudem ist ein solcher USB-Stick auf verschiedenen Ebenen einsetzbar, was bei software-basierten Lösungen oder einer Aktivierung über das Internet nicht der Fall ist. Der Hersteller kann seine Produkte beispielsweise unabhängig von Internet und Betriebssystem schützten. Sicherheitskopien der Software sind dabei aber ohne Probleme möglich. Lizenzinhaber dürfen ihre Software auf fünf verschiedenen Rechnern installieren, ausgeführt werden kann das Programm aber nur in Kombination mit dem entsprechenden Dongle.

Eine der Neuentwicklungen im Bereich der hardware-basierten Kopierschutzlösungen kann sogar für erst noch kommende, zukünftige Versionen eines Programms eingesetzt werden, indem mehrere AES-Schlüssel gespeichert werden. Außerdem lässt sich der Dongle sicher aktualisieren und um neue Features erweitern. So kann der Entwickler verschiedene Spiele-Level und zukünftige Addons schützen oder den Dongle für neue Programme oder Programmversionen verwenden.

Um die Software in den hardware-basierten Kopierschutz einzubauen, bietet sich die automatische Einbindung als einfachste Variante an. Dazu benötigen Sie weder Programmierkenntnisse noch den Sourcecode des zu schützenden Programms, sondern lediglich die kompilierte .exe-Datei. Die Datei wird automatisch komplett verschlüsselt und lässt sich nur noch mit angeschlossenem Dongle ausführen. Ohne Dongle ist die Anwendung damit sicher, da eine AES-Verschlüsselung geknackt werden müsste. Hat der Cracker jedoch den Dongle, kann er die einmal entschlüsselte Anwendung aus dem Speicher auslesen. Der automatische Schutz sollte deshalb eher als Ergänzung zur Einbindung mittels der API betrachtet werden. Bei der Einbindung über die API des Kopierschutzes erhält der Entwickler wesentlich mehr Flexibilität und kann selbst bestimmen, an welchen Stellen im Programm eine Dongle-Abfrage eintreten soll. Voraussetzung ist jedoch, dass er sich vorab mit der Dongle-API beschäftigt und diese möglichst kreativ einsetzt.

Steffen Kätsch

Spore

Ein Dongle ist eine Alternative zu häufig **sehr restriktiven Software-Lösungen**, die leicht den Unwillen der Käufer erregen – wie etwa bei Spore.

Entwicklungsverträge, Teil 13
Wettbewerbsverbote

Ungefähr die Hälfte aller Entwicklungsverträge enthält Wettbewerbsverbote. Wir erläutern, welche Varianten es gibt und wie sie sich auswirken.

Claas Oehler

ist Geschäftsführer Recht beim G.A.M.E.-Bundesverband der Entwickler von Computerspielen.

Neben seiner Tätigkeit für G.A.M.E. arbeitet Claas Oehler auch als Rechtsanwalt für die Kanzlei IHDE & Partner in Berlin und berät Entwickler, Produzenten sowie Verlage.
claas.oehler@onlinelaw.de

Wettbewerbsverbote verlangen von der betroffenen Partei, dem vertragsgegenständlichen Produkt eine Zeit lang keinen Wettbewerb zu machen: Entwicklern wird einstweilen verboten, ähnliche Spiele (»look and feel«) zu entwickeln oder zu vermarkten. Ungleich seltener verlangen Entwickler oder ungeschriebene Vertragspflichten von einem Publisher, dass er während der Vertragslaufzeit keine Produkte vermarktet, die den Absatz des vertragsgegenständlichen Computerspiels mindern könnte. Mehrere Produkte, die zeitgleich eine identische Zielgruppe ansprechen, kannibalisieren sich. Schließlich findet sich eine dritte Gruppe von Wettbewerbsverboten in Entwicklerverträgen: Einige – meistens branchenfremde – Auftraggeber verlangen, dass der Entwickler nicht für unmittelbare Wettbewerber des Auftraggebers tätig wird, weil er Einblick in geheime Geschäfts- und Betriebsabläufe des Auftraggebers erhalten hat, oder weil er nach dem Vertrag auf Kosten des Auftraggebers entwickelte Technologie oder Know-how zurückbehalten darf. Dieser Vorsprung soll den Wettbewerbern nicht zu Gute kommen.

Ausgangslage

Bei erfolgsabhängigen Vergütungen steigt das Interesse an einer optimalen Auswertung: Wenn ein Entwickler ein Produkt, das er vielleicht sogar selbst- oder zumindest mitfinanziert hat, einem Publisher zur Auswertung überlässt und die Parteien eine erfolgsabhängige Vergütung vereinbaren, so hängt die Vergütungshöhe von den Abverkäufen ab. Der Entwickler wird wollen, dass sein Publisher keine unmittelbaren Konkurrenzspiele vermarktet oder vertreibt, die den Absatz seines Produkts und damit seines Gewinnanteils vermindern. Anders als der Entwickler kann der Publisher aber darauf bedacht sein, ein ähnliches Produkt aus seinem Portfolio vorrangig zu vertreiben – sei es, weil das andere Produkt besser ist oder es höhere Ertragschancen verspricht. Er mag Konkurrenzprodukten aus seinem Portfolio motivierter vermarkten, weil dort keine oder geringere Lizenzerlöse an die Entwickler gezahlt werden müssen. Gelegentlich agieren einige Verlage auch strategisch, indem sie potenzielle Wettbewerbsprodukte unter Vertrag nehmen, schlechter vermarkten und so ausbremsen. In der Musikindustrie ist ein solches Vorgehen weit häufiger verbreitet.

Wettbewerbsverbote dürfen aber nicht isoliert gelesen werden. Auch aus anderen vertraglichen Regeln kann sich die Pflicht ergeben, einstweilen keine oder zumindest keine ähnlichen Produkte zu entwickeln oder zu vermarkten.

Nutzungsrechte, Rechteeinräumung

Wenn ein Entwickler sämtliche Rechte an dem Vertragsprodukt in all seinen Bestandteilen (einschließlich der Engine) dem Publisher zur ausschließlichen Verwertung überlässt, also entweder alle Nutzungsrechte für immer oder nur für eine bestimmte Zeit überlässt, dann ist ihm jegliche Eigennutzung der Bestandteile ohnehin untersagt. Die Verwendung einiger Materialien aus dem Computerspiel, der Geschichte oder der Technologie verstößt dann gegen das Urheberrecht. Nur wenn sämtliche Materialien neu erstellt werden und die alten Materialien dabei nicht »durchschimmern«, dann liegt kein Verstoß gegen das Urheberrecht vor. Anders sieht es aber aus, wenn der Entwickler die Rechte an einzelnen Bestandteilen und an seiner Engine behält – und das ist praktisch die Regel. Hier verstößt er nicht gegen das Urheberrecht, wenn er diese Elemente für andere Produkte einsetzt. Ob er dann für Wettbewerber tätig werden darf, ist letztlich eine Frage des Einzelfalls und des Preises.

First Look Klauseln

Auch First-Look- und Last-Refusal-Klauseln (siehe /GameStar/dev 2/2008) können faktisch dazu führen, dass eine Tätigkeit für andere Auftraggeber ausgeschlossen ist. Sämtliche Produkte, die ein Entwickler in den dem Vertragsschluss folgenden Jahren entwickelt, muss er zunächst dem Publisher anbieten. Ist dann zusätzlich ein Wettbewerbsverbot vereinbart, werden die Klauseln wohl im Zusammenhang auszulegen sein. Das heißt bei Freigabe eines Produkts nach der Option wird, soweit die Klausel nichts Gegenteiliges sagt, der Entwickler nicht gegen das Wettbewerbsverbot verstoßen. Andernfalls wären beide Klauseln im Zusammenhang (Juristen nennen das auch Summierungseffekt) unter Umständen unwirksam, weil sich der Entwickler faktisch überhaupt nicht mehr wirtschaftlich betätigen kann.

Key-Personal Klauseln

Das Verbot, Mitarbeiter in anderen Projekten einzusetzen, wurde in einem früheren Beitrag

behandelt. Solche Mitarbeiterbindungsklauseln führen faktisch auch dazu, dass der Entwickler seine Mitarbeiter während der Entwicklung der Auftragsproduktion nicht dazu einsetzen kann, andere Produkte zu entwickeln. Faktisch bedeutet das für viele Studios ein Wettbewerbsverbot, dass nur durch die Gründung von Schwestergesellschaften umgangen werden kann.

Wettbewerbsverbote

Gesetzliche Wettbewerbsverbote werden in Entwicklungsverträgen selten relevant. Zum Beispiel schreibt das Verlagsgesetz in §2 vor: »Der Verfasser hat sich während der Dauer des Vertragsverhältnisses jeder Vervielfältigung und Verbreitung des Werkes zu enthalten, die einem Dritten während der Dauer des Urheberrechts untersagt ist. Dem Verfasser verbleibt jedoch die Befugnis zur Verbreitung:

1. für die Übersetzung in eine andere Sprache oder in eine andere Mundart;
2. für die Wiedergabe einer Erzählung in dramatischer Form oder eines Bühnenwerkes in der Form einer Erzählung;
3. für die Bearbeitung eines Werkes der Tonkunst, soweit sie nicht bloß ein Auszug oder eine Übertragung in eine andere Tonart oder Stimmlage ist;
4. für die Benutzung des Werkes zum Zwecke der mechanischen Wiedergabe für das Gehör;
5. für die Benutzung eines Schriftwerkes oder einer Abbildung zu einer bildlichen Darstellung, welche das Originalwerk seinem Inhalt nach im Wege der Kinematographie oder eines ihr ähnlichen Verfahrens wiedergibt.

Auch ist der Verfasser zur Vervielfältigung und Verbreitung in einer Gesamtausgabe befugt, wenn seit dem Ablaufe des Kalenderjahres, in welchem das Werk erschienen ist, zwanzig Jahre verstrichen sind.«

Während sich beim Verlagsvertrag über geschützte Werke der Verfasser nach § 2 Abs 1 während der ganzen Dauer des Vertragsverhältnisses der Vervielfältigung und Verbreitung seines Werkes zu enthalten hat, ist diese Enthaltungspflicht beim Verlagsvertrag über gemeinfreie Werke auf die Dauer von sechs Monaten nach dem Erscheinen begrenzt. Dieses Gesetz stellt letztlich nur klar, was Rechtsfolge der Einräumung von ausschließlichen (exklusiven) Nutzungsrechten ist und lässt einzelne Nutzungen ausnahmsweise zu. Weitere Verbote finden sich z.B. in den §§ 74 ff, 112 HGB für Handelsvertreter und Gesellschafter.

Ungeschriebene Wettbewerbsverbote

Enthaltung von Wettbewerbstätigkeiten gibt es nicht. Dennoch gibt es Fallkonstellationen des Leistungsaustausches, in denen es aus sachlichen Erwägungen geboten ist, auch ohne ausdrückliche Vereinbarung der Parteien ein Wettbewerbsverbot aus dem Vertragszweck abzuleiten. Während der Dauer eines Lizenzvertrages darf sich niemand so verhalten, dass die Erfüllung des Vertragszwecks gefährdet wird. Aus diesem allgemeinen Grundsatz lassen sich ungeschriebene Wettbewerbsverbote begründen. Unbillig wäre es, wenn der Entwickler kurz nach der Fertigstellung einer Auftragsproduktion ein ähnliches, fast identisches Spiel auf den Markt brächte. Als Kriterien können der Projektumfang, die finanziellen Aufwendungen sowie die übrigen Verwertungsmodalitäten herangezogen werden. Als Faustregel kann man sich wohl merken: Wer etwas zum Vermarkten »verkauft«, darf eine Zeitlang keine fast identische Leistung (und erst recht nicht unter Nutzung identischer Materialien) anbieten, die die Vermarktung vereiteln würde.

Grenzen von Wettbewerbsverboten

Wettbewerbsverbote sind nicht grenzenlos zulässig. Grundsätzlich verbietet das Kartellrecht (Artikel 81, 82 EGV, § 1 GWB) Abreden, die den Wettbewerb beschränken. Nur in engen Grenzen sind Wettbewerbsverbote erlaubt. So gibt es etwa vertragsimmanente Ausnahmen und partielle gesetzliche Regelungen (z. B. in § 2, 39 Satz 2 VerlagsG), in Gesellschaften (z. B. § 113 HGB) oder für Handelsvertreter und entsprechend für Arbeitnehmer (§ 110 Gewerbeordnung, § 90a HGB).

Wettbewerbsverbote, die nicht unter solche Sondervorschriften fallen, sind dennoch nicht per se unwirksam. Vielmehr müssen sie auch geeignet sein, eine künstliche Veränderung der Marktbedingung zu bewirken oder zu bezwecken und damit zu einem Verlust an autonomer oder individueller Selbstbestimmung der Marktteilnehmer führen. Dabei muss es sich aber um eine spürbare Beeinträchtigung handeln. Daran fehlt es, wenn die beteiligten Unternehmen addiert nicht mehr als 15% der Marktanteile ausmachen. Wenn ein Unternehmen oder Konzern weniger als 250 Mitarbeiter beschäftigt und weniger als 40 Millionen im Jahr umsetzt und die Bilanzsumme weniger als 27 Millionen aufweist, kann auch davon ausgegangen werden, dass dieses Unternehmen nicht in der Lage ist, den Handel spürbar zu beeinträchtigen. Daran fehlt es jedenfalls dann, wenn die Unternehmen je nach Ausgestaltung und Gegenstand des Vertrages zwischen 15 und 30 % der Marktanteile nicht überschreiten. Für die meisten Entwicklerverträge haben diese Vorschriften deshalb weniger Bedeutung. Im Übrigen gelten die nachfolgend beschriebenen Regeln sinngemäß auch im Kartellrecht.

AGB Inhaltskontrolle

Häufiger wird die Inhaltskontrolle von Allgemeinen Geschäftsbedingungen relevant. Verträge werden oft als Musterverträge einseitig gestellt. Sie sind deshalb oft als Allgemeine Geschäftsbedingungen nach besonderen Regeln (§ 305 f. BGB) zu kontrollieren. Auch wenn der Vertrag im Übrigen verhandelt wurde, können einzelne Klauseln diesen Regeln unterliegen, wenn Sie nicht zur Disposition bei den Vertragshandlungen standen. Bei Wettbewerbsverboten ist das nicht selten der Fall. Die Prüfung kann nur

Typische Vertragsinhalte

0. Vorspann und wesentliche Punkte, Zusammenfassung

1. Definitionen

2. Entwicklungsprozess
- Leistungsabschnitte, Termine
- Kommunikation (Ansprechpartner, Weisungen)
- Teil-/Abnahmeverfahren
- Inhalte, die vom Publisher geliefert werden
- Einbau
- Development Kits
- Änderungen
- Verzögerungen und deren Folgen
- Bindung des Kernteams
- Lokalisierungen, Service, Support (Versicherungen)

3. Rechteübertragung und Verwertung
- Rechteeinräumung (Lizenz), Gebiet, Dauer
- Exklusivität (in US-Verträgen oft »work made for hire« und Urheberschaft)
- Verwertungsrechte, Titelrechte, Marken- und Kennzeichen
- Charaktere, Dokumentation, Quellcode
- Ausnahmen und Vorbehalte von der Rechteeinräumung
- Credits, Anbringen von Logos und Urheberbenennungen
- Optionen, Nachfolger
- Wettbewerbsverbote

4. Finanzierung, Vergütung, Beteiligung
- Vorauszahlungen
- Abrechnungsschlüssel
- Royalties, Verrechnungen
- Beteiligung an anderen Produkten
- Konzernverrechnung, Kompensationsgeschäfte
- Zahlungswege, Bankverbindung
- Retourreserven
- (Doppel-)Besteuerung

5. Pflichten der Parteien
- Entwicklerpflichten
- Publisherpflichten
- Vertrieb, Marketing

6. Laufzeit und Kündigung
- »Kündbarkeit«:
  - vor Fertigstellung, nach Fertigstellung
  - wegen nicht erreichter Konsolenhersteller-anforderung
- Pflichtverletzungen, Abmahnungen, Fristen, Folgen, Ausgleichszahlungen, Rechterückfall, Insolvenzfall und Sicherungsrechte
- Fortwirkung von Klauseln

7. Garantien, Freistellungen, Haftungsbeschränkungen
- Fehlerklassen, Rechteeinräumung
- Mitverschulden
- Mitwirkung an Prozessen

8. Vertraulichkeit
- Vertrauliche Informationen, Ausnahmen
- Verpflichtung der Mitarbeiter

9. Allgemeine Klauseln
- Abtretbarkeit, Aufrechnung
- Erklärungen, Zugang
- Schiedsklausel
- Anwendbares Recht, Gerichtsstand
- Vollständige salvatorische Klausel

Vertragsmuster: http://contracts.corporate.findlaw.com/type/operations/ip.html

Beispiel einer Klausel:

§ 1 Umfang des Wettbewerbsverbotes
I. Freelancer verpflichtet sich, während eines bestehenden Dienstvertrages und während der Dauer von .zwei (2) Jahren nach Beendigung des Dienstverhältnisses im Falle der Begründung eines eigenen Unternehmens nicht für die nachstehend genannten und solche Auftraggeber tätig zu werden, die in den letzten zwei Jahren vor Beendigung des Arbeits- oder Dienstverhältnisses Auftraggeber bzw. Kunden der Auftraggeber GmbH waren. (Liste der Kunden)
II. Geht Freelancer ein anderes Dienst- oder Arbeitsverhältnis ein oder tritt er in ein anderes Unternehmen/Gesellschaft ein, so verpflichtet er sich, auf Kunden von Auftraggeber nicht einzuwirken, diesen aus bestehenden Verträgen zu verdrängen oder während der Dauer des Verbotes neue Beratungs-/Projekt-/Werk- oder andere Verträge im Tätigkeitsbereich von Auftraggeber nicht mehr mit Auftraggeber sondern Dritten oder dem neuen Unternehmen abzuschließen. Diese Verpflichtung gilt entsprechend für Vorstandsmitglieder, Geschäftsführer oder persönlich haftende Gesellschafter.
III. Freelancer verpflichtet sich ferner, während der Dauer von .zwei (2) Jahren nach Beendigung des Arbeits- oder Dienstverhältnisses keine Mitarbeiter oder Lieferanten von Auftraggeber abzuwerben. Die unter I und II getroffenen Wettbewerbsverbote gelten entsprechend. Zu den Mitarbeiter und Lieferanten zählen insbesondere folgende Personen: ...
§ 2 Karenzentschädigung
I. Auftraggeber verpflichtet sich, Freelancer für die Dauer des Wettbewerbsverbotes eine Entschädigung zu zahlen, die für jedes Jahr des Verbots mindestens die Hälfte der von Freelancer zuletzt bezogenen vertragsmäßigen Leistungen erreicht.
II. Die Karenzentschädigung wird fällig am Schluss eines jeden Monats.
§ 3 Anrechnung anderweitigen Verdienstes
I. Auf die Karenzentschädigung wird alles angerechnet, was Freelancer durch anderweitige Verwertung seiner Arbeitskraft erwirbt oder zu erwerben böswillig unterlässt, soweit die Entschädigung unter Hinzurechnung dieses Betrages den Betrag der zuletzt von ihm bezogenen vertragsmäßigen Leistungen um mehr als 1/10 übersteigen würde. Ist Freelancer durch das Wettbewerbsverbot gezwungen worden, seinen Wohnsitz zu verlegen, so tritt an die Stelle des Betrages von einem Zehntel der Betrag von einem Viertel. § 74c HGB gilt entsprechend.
II. Freelancer verpflichtet sich, während der Dauer des Wettbewerbsverbots (§1) auf Verlangen Auskunft über die Höhe seiner Bezüge zu geben und die Anschrift seines jeweiligen Arbeitgebers mitzuteilen. Am Schluss eines Kalenderjahres ist er verpflichtet, seine Lohnsteuerkarte vorzulegen.

im Einzelfall erfolgen. Faustregel: Je strenger das Verbot, desto eher ist die Klausel unwirksam.

In gegenständlicher Hinsicht darf sich das Wettbewerbsverbot nur auf Titel beziehen, die dem vertraglich Vereinbarten nach Art und Umfang entsprechen, wobei Maßstab für diese Kriterien das konkret vereinbarte Spielkonzept ist. Nur im Hinblick auf dieses droht den Parteien aufgrund ihrer vertraglichen Nähe eine Gefahr durch eine Parallelentwicklung und -verwertung. Eine inhaltlich weiter gefasste Klausel, die sich beispielsweise nur auf das Genre bezieht, wäre wohl nicht mehr geboten. Hier müssen weitere Gründe hinzutreten, um die Klausel zu rechtfertigen. Die zeitliche Komponente eines Wettbewerbsverbotes ist im Wesentlichen von den Prognosen der Ertragsentwicklung und der konkreten Ausgestaltung der Auswertungsphasen abhängig. AAA-Titel sprechen eine technik- und innovationsaffine Zielgruppe an, sodass insbesondere dem ersten Jahr in der Auswertungsphase eine entscheidende Rolle für den wirtschaftlichen Erfolg zukommt. Für die späteren Auswertungsphasen (Zweit- und Drittverwertung, z.B. im OEM-Sektor) ist der Markterfolg nach dem unmittelbaren Erscheinen relevant. Auf diesem Markt tritt der Titel dann aufgrund der aus technischer Sicht veralteten Umsetzung nicht mehr in Wettbewerb zu neuen AAA-Titeln. Ein Wettbewerbsverbot wird in der Regel das erste Jahr nach dem Erscheinen nicht deutlich (ohne weitere Gründe) überschreiten dürfen. Auch wenn bei Casual Games die Erträge aufgrund deutlich niedrigerer Preise geringer sind, ist zu berücksichtigen, dass hier die technische Realisation keine relevante Rolle spielt, da fast immer die zeitlose Spielidee im Vordergrund steht, sodass eine wirtschaftliche Auswertung deutlich länger möglich ist.

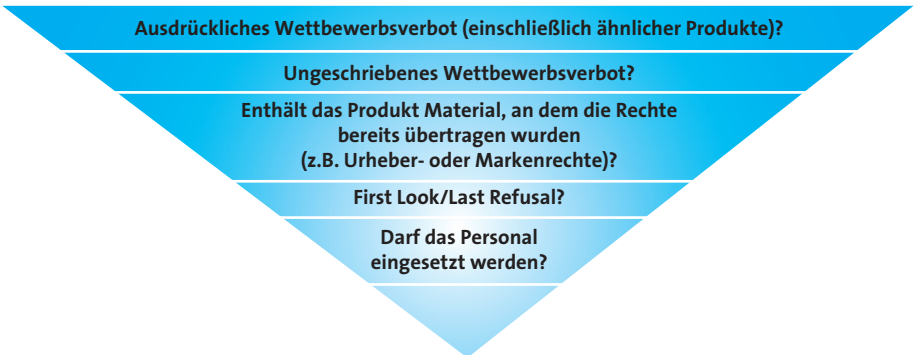
Außerhalb von Standardverträgen bestimmen die Vertragspartner grundsätzlich selbst, wie ihre gegenläufigen Interessen angemessen auszugleichen sind. Der zum Ausdruck gebrachte übereinstimmende Wille der Vertragsparteien lässt deshalb in der Regel auf einen sachgerechten Interessenausgleich schließen. Das setzt aber voraus, dass eine wechselseitige Einflussnahme und Inhaltsbestimmung tatsächlich möglich ist. Das BVerfG betont immer wieder, dass bei ungleichen Verhandlungspositionen in einem Vertragsverhältnis, in dem ein Partner ein solches Gewicht hat, dass er den Vertragsinhalt faktisch einseitig bestimmen kann, es Aufgabe des Rechts ist, zu verhindern, dass sich für einen Vertragsteil die Selbstbestimmung in eine Fremdbestimmung verkehrt. Solche Verträge unterliegen dann häufig dem Urteil der Sittenwidrigkeit.

Sonderfälle Arbeitnehmer

Für Arbeitnehmer gelten die besonderen Regeln zu Wettbewerbsverboten, die auch für kaufmännische Hilfspersonen und Handelsvertreter gelten (§ 110 GeWO). Wegen des vergleichbaren Schutzbedürfnisses wendet die Rechtsprechung diese Regeln auch auf formal Selbstständige, aber wirtschaftlich abhängige freie Mitarbeiter an (arbeitnehmerähnliche Personen).

Durch die Wettbewerbsabrede kann dem Arbeitnehmer (oder Freelancer) jede Tätigkeit in einem Konkurrenzunternehmen verboten sein (unternehmensbezogene Konkurrenzklausel). Die Wettbewerbsabrede kann sich aber auch auf das Verbot nur bestimmter Tätigkeiten in einem Konkurrenzunternehmen beschränken (partielle Konkurrenzklausel). Wirksam sind solche Abreden aber nur, wenn die Wettbewerbsabrede gemäß § 74 I HGB schriftlich vereinbart ist (§ 126 BGB). Zusätzlich muss die Urkunde, die sämtliche Abreden enthält, dem Arbeitnehmer ausgehändigt werden. Bei Nichteinhaltung der Formvorschriften ist die Abrede gemäß § 125 S. 1 BGB nichtig. Ferner müssen die Ausgleichsvorschriften eingehalten werden. Inhaltlich muss ein berechtigtes Interesse des Arbeitgebers bestehen, das Verbot darf zwei Jahre nicht übersteigen, und es muss eine ausreichende Karenzentschädigung gezahlt werden. Einzelheiten dazu regeln die § 74 ff HGB.

Werden diese Grenzen überschritten, so unterscheidet das Gesetz im Hinblick auf die Rechtsfolgen zwischen Nichtigkeit und Unverbindlichkeit. Ist ein Wettbewerbsverbot unverbindlich, steht dem Freelancer oder Arbeitnehmer ein Wahlrecht zu. Er kann sich für die Einhaltung des Wettbewerbsverbots entscheiden – mit der Folge, dass ihm auch die Karenzentschädigung zusteht oder sich aber auch von dem Wettbewerbsverbot lösen. In diesem Fall verliert er zugleich seinen Anspruch auf die Karenzentschädigung. Der Arbeitgeber kann sich nicht auf das Verbot berufen. **Claas Oehler**


Ausdrückliches Wettbewerbsverbot (einschließlich ähnlicher Produkte)?
Ungeschriebenes Wettbewerbsverbot?
Enthält das Produkt Material, an dem die Rechte bereits übertragen wurden (z.B. Urheber- oder Markenrechte)?
First Look/Last Refusal?
Darf das Personal eingesetzt werden?

Mehr Spaß im Job

Wir brauchen dich als:

- Gamedesigner (m/w)
- Interface-Designer (m/w)
- Senior Flash Entwickler (m/w)
- Technischer Projektleiter (m/w)

TRAVIANGAMES

traviangames.com/jobs

Medienrecht
IT-Recht
Forderungseinziehung

Dr. Alexander Wachs
Rechtsanwalt

Die Kanzlei Dr. Wachs berät Sie gern u.a. bei Fragen zu den Themenkreisen:

- Urheberrechte und Nutzungsrechte
- Lizenverträgen
- Softwareverträge - Vertragsgestaltung
- Hardwarekaufvertrag
- Software- / Hardwarepflegevertrag
- Verträge und Allgemeine Geschäftsbedingungen im E-Business
- Datenschutz und Datensicherheit
- Webseiten Check
- Domainrecht
- E-Commerce

www.dr-wachs.de
Telefon +49 (0)40-44 46 55 08
Telefax +49 (0)40-44 46 55 10

»Ich will immer etwas Neues erschaffen.«

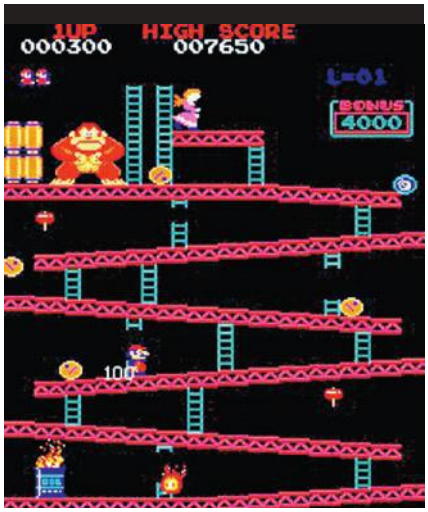
Im Gespräch mit Shigeru Miyamoto

Er ist ein Genie, er ist unglaublich sympathisch, und er hat klare Ansichten: Die Designer-Legende über ihre Anfänge, aktuelle Projekte, die Kritik an Casual Games und die Zukunft von Nintendo.

Shigeru Miyamoto

ist General Manager bei Nintendo.

Miyamotos Karriere begann 1981, als ihn die kriselnde Firma Nintendo beauftragte, Figuren für einen Spielautomaten zu entwickeln. Er entwarf Donkey Kong und Super Mario, der damals noch Jumpman hieß. Der Rest ist Videospielgeschichte. Shigeru Miyamoto ist dafür bekannt, dass er sich für seine Spielideen vor allem vom Alltagsleben inspirieren lässt. Die Idee zu Wii Fit entstand etwa, als er für sich feststellte, wie viel Spaß ihm das morgendliche Wiegen macht.



»Meine Aufgabe bestand bei Donkey Kong in erster Linie darin, Konzeptzeichnungen anzufertigen – mit Stift und Papier.«

Making Games Nach Super Smash Bros. Brawl melden Sie sich nun als Produzent von Wii Music zurück. Das zentrale Thema bei Wii Music ist – wie der Name schon sagt – Musik. Welchen Stellenwert hat Musik in Ihrem Leben? Spielt Musik eine wichtige Rolle für Sie?

Shigeru Miyamoto Nun, ich würde sagen, dass Musik ein durchaus wichtiges Thema für mich ist. Vor über 20 Jahren habe ich damit angefangen, unterschiedliche Musikinstrumente zu spielen. Ja tatsächlich – ich spielte sogar in verschiedenen Bands. Zu diesen Leuten habe ich auch heute noch Kontakt. Durch die Musik sind Freundschaften entstanden, die ich nach wie vor pflege. Diese Tatsache ist einer der Gründe, weshalb mir die Musik sehr viel bedeutet. In meinem Büro und zu Hause in meinem Wohnzimmer stehen sogar einige Gitarren rum. Wenn ich Muße habe, schnappe ich mir eine Klampfe und jamme ein bisschen vor mich hin. Ich komme dann auf andere Gedanken, es gibt mir ein sehr wohliges Gefühl. Es ist mir auch sehr wichtig, dass sich meine Kinder mit Musik befassen. Mein Sohn spielt Gitarre und Schlagzeug, meine Tochter nimmt Trompeten-Unterricht und spielt E-Gitarre in einer Band. Deshalb ist Musik eines der wichtigsten Gesprächsthemen in meiner Familie.

Making Games Lassen Sie uns über Ihr aktuelles Projekt Wii Music reden. Ein Spiel, das ähnlich wie Guitar Hero oder Rockband in die Musik-Kerbe schlägt. Was macht Wii Music anders im Vergleich zu den eben genannten Spielen?

Shigeru Miyamoto Richtig – das Hauptelement von Wii Music ist natürlich die Musik. Jedoch ist der Titel nicht mit den Spielen vergleichbar, die Sie eben erwähnt haben. Viele Leute, die zum ersten Mal von Wii Music hören, denken vielleicht, dass nun auch Nintendo auf den Standard-Musikzug aufgesprungen ist. Wenn sich diese Leute jedoch erst einmal mit Wii Music beschäftigt haben, werden sie schnell bemerken, dass sie mit ihrer Vermutung völlig falsch lagen. Bei anderen Genre-Vertretern muss der Spieler im richtigen Moment einen bestimmten Knopf drücken. Das ist das Hauptspielelement. Die Herausforderung besteht in erster Linie darin, die vorgegebenen Tastenreihenfolgen so präzise wie möglich zu absolvieren, um ein gutes Ergebnis zu erzielen. Genau hier liegt der Unterschied zu Wii Music. Bei Wii Music kann der Spieler die Lieder so einspielen, wie es ihm gefällt. Er muss sich nicht an einen vorgegebenen Rhythmus halten. Wir lassen dem Spieler alle Freiräume, zu improvisieren. Der Reiz liegt vor allem darin, einen Song in unzähligen unterschiedlichen Varianten zu spielen. Man könnte Wii Music mit einer Jam-Session vergleichen. Zu Beginn der Session steht noch nicht fest, wie das Endresultat klingen wird. Der Spieler muss sich an keine Vorgaben halten.

Making Games Welche Zielgruppe möchten Sie mit Wii Music ansprechen? Richtet sich der Titel an Leute mit musikalischen Vorkenntnissen oder eher an jene, die bislang noch keinen Kontakt mit dem Thema Musik hatten?

Shigeru Miyamoto Wii Music ist ein Titel für jedermann. Es spielt keine Rolle, ob der Spieler musikalisch ist oder nicht. Nein, es ist sogar egal, ob derjenige schon einmal ein Videospiel gespielt hat. Mit Wii Music lernt der Spieler den Umgang mit einem Musikinstrument. Außerdem macht er die wertvolle Erfahrung, welchen Spaß es macht, zu musizieren. Ich hoffe inständig, dass viele diese Erfahrung machen werden. Ich habe bereits mit vielen unterschiedlichen Leuten Wii Music gespielt – auch mit professionellen Musikern. Besonders für musik-affine Menschen stellt Wii Music eine große Herausforderung dar. Beim Umgang mit den Instrumenten sammeln speziell Musiker neue, wertvolle Erfahrungen. Das finde ich wirklich sehr interessant. Plötzlich befinden sich Musiker auf der gleichen Ebene wie all jene Leute, die noch nie ein Instrument in den Händen hielten. Wii Music verbindet diese beiden Parteien und beweist eindrucksvoll, dass eine musikalische Konstellation aus Musikern und Amateuren funktionieren kann. Die Möglichkeit, eine solche ungewöhnliche Kombination zu realisieren, ist eine der Hauptstärken von Wii Music.

Making Games Miyamoto-san, Sie haben es eben selber gesagt: Wii Music ist ein Spiel für je-

dermann. Mit anderen Worten handelt es sich also um ein weiteres Casual Game. Wann erscheint endlich ein neues Core Game für Nintendo Wii ? Das nächste Super Mario-Abenteuer oder vielleicht ein völlig neues Kid Icarus-Spiel. Was möchten Sie all den Leuten sagen, die auf ein neues »richtiges« Spiel warten?
Shigeru Miyamoto (schmunzelt) Nintendo wird definitiv neue Spiele ankündigen. Wir haben beschlossen, zunächst einige Titel, die sich derzeit in der Produktion befinden, fertig zu stellen. Sobald diese Arbeiten beendet sind, widmen wir uns wieder verstärkt der Konzeption neuer, »richtiger« Spiele. Denn genau in der Erschaffung solcher Titel liegt die Stärke von Nintendo. Im Moment kann ich Ihnen jedoch noch keine genauen Termine oder Details zu kommenden Spielen sagen. Aber Sie dürfen sich auf zukünftige Ankündigungen freuen. Welche Spiele wir veröffentlichen werden, ist bislang noch unklar. Aber wer weiß, vielleicht können wir heute oder morgen tatsächlich ein neues Kid Icarus ankündigen (lacht).

Making Games Lassen Sie uns über Ihre Anfänge in der Branche reden. Man könnte Sie als klassischen Quereinsteiger bezeichnen. Welche Berufsvorstellungen hatten Sie in Ihrer Kindheit?
Shigeru Miyamoto Ursprünglich habe ich mit dem Gedanken gespielt, eine Karriere als Mangaka (Manga-Zeichner) einzuschlagen. Ich hatte aber auch noch einen anderen Plan: Ich wollte Puppentheater-Marionetten entwerfen.

Making Games Mit dem Aracade-Spiel Donkey Kong haben Sie schlagartig den Durchbruch in der Spiele-Branche geschafft. Wie lief damals die Entwicklung ab?
Shigeru Miyamoto Unser damaliges Team bestand aus fünf Mitarbeitern, darunter auch ein Programmierer. Ich selber war zu diesem Zeitpunkt nur sehr oberflächlich mit der Technik vertraut. Meine Erfahrungen und das Wissen, das ich während meiner langjährigen Laufbahn gesammelt habe, waren damals noch sehr beschränkt. Meine Aufgabe bestand in erster Linie darin, Konzeptzeichnungen anzufertigen – mit Stift und Papier. Die Computer, mit denen wir damals gearbeitet haben, waren riesige Maschinen – kein Vergleich zur heutigen Technik. Wir arbeiteten mit der so genannten Floppy Disk, ein Datenträger, der sich zu jener Zeit gerade in der Branche etablierte.

Making Games Vermissen Sie diese Pionierzeiten, oder sind Sie mit dem heutigen Stand der Technik glücklicher?
Shigeru Miyamoto Beides hat seine Vor-und Nachteile. Was schlussendlich zählt, ist die Tatsache, dass ich schon immer mit Leuten zusammenarbeiten durfte, die das Beste aus dem machen, was ihnen zur Verfügung steht. Man muss stets das volle Potenzial der technischen Möglichkeiten ausschöpfen. Diesem Vorsatz wollte ich immer gerecht werden. Auch wenn es ab und zu sehr schwierig war, hat es mir jederzeit großen Spaß gemacht. Aufgrund der stetigen Entwicklung hat man natürlich die Möglichkeit, völlig neue Dinge zu erschaffen. Dinge, die früher in dieser Form niemals möglich gewesen wären. Aber wie ich bereits eingangs sagte: Jede Zeit ist auf ihre Art eine tolle Zeit.

Jede Zeit ist auf ihre Art eine tolle Zeit.

Making Games Was halten Sie von der derzeitigen Kreativität in der Spiele-Industrie?
Shigeru Miyamoto Das ist eine schwierige Frage, die ich Ihnen so nicht beantworten kann. Ich bin nicht in der Position, die Arbeit und die Kreativität anderer Leute in der Branche zu beurteilen. Die Videospiel-Industrie wächst kontinuierlich. Jedoch ist die Herangehensweise einiger »Newcomer« eine völlig andere geworden, wie sie noch zu meiner Zeit üblich war. Viele Entwickler stellen Spiele her, weil sie selber gerne spielen. Der Videospielmarkt ist inzwischen extrem trendabhängig. Wenn sich beispielsweise ein Genre von heute auf morgen als erfolgreich herausstellt, springt die Industrie auf diesen Zug auf. Plötzlich veröffentlicht jedes Entwicklerstudio einen ähnlichen Titel. Ein weiteres Beispiel: Wenn eine Firma mit einem Spiel große Erfolge verzeichnet, kann man sicher davon ausgehen, dass nur kurze Zeit nach der Veröffentlichung eine Fortsetzung angekündigt

Wer weiß, vielleicht können wir heute oder morgen tatsächlich ein neues Kid Icarus ankündigen.

Shigeru Miyamoto beim Interview in London: »Wir haben beschlossen, zunächst einige Titel, die sich derzeit in der Produktion befinden, fertig zu stellen. Sobald diese Arbeiten beendet sind, widmen wir uns wieder verstärkt der Konzeption neuer, ›richtiger‹ Spiele.«

wird. Wenn jeder das Gleiche macht, schadet das meiner Meinung nach der Industrie. Für mich persönlich ist es immer sehr wichtig, etwas völlig Neues zu erschaffen. Ich würde mir wünschen, dass andere Software-Entwickler genauso denken.

» Wir lassen dem Spieler alle Freiräume, zu improvisieren. «

Making Games Apropos Kreativität: Was inspiriert Sie? Woher nehmen Sie die Ideen?
Shigeru Miyamoto Diese Frage wurde mir schon oft gestellt und es fällt mir jedes Mal sehr schwer, eine konkrete Antwort darauf zu geben. Wenn ich genau wüsste, woher ich neue Ideen beziehen kann, würde ich diese Quelle der Inspiration natürlich ständig aufs Neue aufsuchen, um neue Projekte zu entwickeln. Ich glaube in erster Linie kommt es auf meinen Gemütszustand an. Aus Situationen, in denen ich traurig, nachdenklich oder glücklich bin, resultieren oftmals – unbewusst – großartige Ideen. Solche Geistesblitze habe ich ab und zu morgens im Badezimmer, in einer Mitarbeiter-Konferenz oder wenn ich durch ein Museum schlendere. Hin und wieder kommen mir auch beim Fernsehen urplötzlich gute Einfälle.

» Wii Music beweist, dass eine musikalische Konstellation aus Musikern und Amateuren funktionieren kann. «

Making Games Stichwort TV: Waren Sie an der Super Mario Film-Adaption beteiligt?
Shigeru Miyamoto Nein, ich war nicht wirklich in das Projekt involviert. Als Erfinder von Mario bat man mich jedoch darum, meine Meinung zur ersten Vorab-Version des Films zu äußern.

Making Games Können wir uns auf einen weitern Film freuen, an dem Sie direkt beteiligt sind?
Shigeru Miyamoto Darüber habe ich, ehrlich gesagt, noch nie wirklich nachgedacht. Ich bin nun mal in erster Linie Spiele-Entwickler. In Bezug auf den damaligen Super Mario-Kinofilm haben mich jedoch immer wieder Leute gefragt, warum ich nicht selbst auf dem Regiestuhl Platz genommen habe. Einige Leute glauben, dass sich die Film- und Videospiel-Industrie sehr ähneln und es ein Leichtes für mich wäre, einen Film zu drehen. Das stimmt jedoch nicht. Meiner Meinung nach sind Videospiel- und Filmproduktion zwei völlig unterschiedliche Dinge. Wer weiß, vielleicht wird es irgendwann tatsächlich einen neuen Super Mario-Film geben. In diesem Fall würde ich vielleicht im Hintergrund die Fäden ziehen, den Job als Regisseur würde ich jedoch in Auftrag geben.

Making Games Dann lassen Sie uns nun wieder über Ihre Arbeit reden. Gibt es ein Projekt, auf das Sie besonders stolz sind?
Shigeru Miyamoto Schwierig ... Natürlich war Donkey Kong ein extrem wichtiges Projekt für meine Laufbahn. Deshalb bin ich auch heute noch sehr stolz auf den Automaten. Donkey Kong war schlussendlich zu verdanken, dass ich heute diese Position habe. Aber auch Super Mario Bros. muss in diesem Zusammenhang erwähnt werden. Bei diesem Spiel habe ich so viel Unterstützung und Feedback von Spielern aus der ganzen Welt bekommen. Was mich jedoch besonders stolz macht, ist die Tatsache, dass ich schon viele Jahre mit dem gleichen Team arbeite. Das Super Mario Bros.-Team bestand lediglich aus drei Leuten. Im Laufe der Jahre wurde das Team kontinuierlich größer und Dutzende der Leute arbeiten schon seit langer Zeit für Nintendo. Es ist also weniger ein spezielles Projekt oder ein bestimmtes Spiel, das mich stolz macht, sondern vielmehr dieser Team-Aspekt.

Making Games Welches Videospiel, das Sie zuletzt privat gespielt haben, hat Sie beeindruckt?
Shigeru Miyamoto Mein derzeitiger Hit ist Rhythm Heaven. Man könnte das Spiel zwar als Wii Music-Rivalen einstufen, aber Rhythm Heaven macht mir einfach sehr viel Spaß.

Making Games Haben Sie einen favorisierten Videospiel-Charakter?
Shigeru Miyamoto (lacht) Ja natürlich: Mario! Und vielleicht noch Bowser.

Making Games Wie sieht die Zukunft von Shigeru Miyamoto und Nintendo aus?
Shigeru Miyamoto Ich habe in der Vergangenheit mehrmals mit dem Gedanken gespielt, die Arbeit als Videospiel-Designer an den Nagel zu hängen. Es gab Momente, in denen ich einfach ausgelaugt und müde war. Heute hege ich jedoch mehr denn je den Wunsch, auch in Zukunft neue Videospiele zu entwickeln. Ich werde nicht aufgeben und auch weiterhin eifrig meiner Arbeit nachgehen. Ich würde unheimlich gerne ein neues Zelda-Spiel entwickeln. Gleichzeitig habe ich aber auch den Wunsch, etwas völlig Neues zu erschaffen. Etwas Unverbrauchtes und noch nie Dagewesenes. Nintendo teilt diese Meinung aus der Sicht eines Unternehmens mit mir. Auch der ehemalige Nintendo-Präsident Hiroshi Yamauchi hegte stets diesen Wunsch. Der Auftrag von Nintendo ist es, Leute zu unterhalten. Und genau das haben wir uns auch für die Zukunft auf die Fahnen geschrieben.

Nino Kerl

»Wii Music ist ein Titel für jedermann. Es spielt keine Rolle, ob der Spieler musikalisch ist oder nicht. Nein, es ist sogar egal, ob derjenige schon einmal ein Videospiel gespielt hat.«

TRINIGY
www.trinigy.net

"We're hiring!"

- Senior Engine Engineer
- Support Engineer

Vision Engine 7 - leading technology for leading platforms

XBOX 360 · XBOX LIVE arcade · PLAYSTATION Network · Tools & Middleware PLAYSTATION 3 · Wii · Games for Windows

www.trinigy.net

Kollisionserkennung als Gameplay-Feature
Swordmaster

Das fünfköpfige Studententeam RedFingerArts baute nach dreijährigem Studium ein aufwändiges Schwertkampf-Multiplayer-Spiel.

Rainer Lonau

ist Junior Programmierer bei GPI.

Rainer schloss im September 2008 erfolgreich sein Studium »Gamedesign B.Sc.« an der Mediadesign Hochschule ab. Er übernahm im Projekt Swordmaster die technische Leitung und die Programmierung. Zusammen mit drei weiteren Teammitgliedern der Studentengruppe RedFingerArts arbeitet er seit Oktober für GPI an einem unangekündigten Titel.
lonau.rainer@drei-c-gmbh.de

Markus Ambrus

studierte Gamedesign an der Mediadesign Hochschule München.

Markus war bei Swordmaster für das Charakter-Modellierung, die Animationen und das Projektmanagement zuständig.
markus.ambrus@googlemail.com

Beim Projekt Swordmaster handelt es sich um ein innovatives Multiplayer-Schwertkampf-Spiel. Zwei Kontrahenten kämpfen in einem Arena- Level gegeneinander, während weitere Spieler im Zuschauermodus den Kampf verfolgen und sogar beeinflussen können. Im Gegensatz zu anderen Spielen dieser Art wird bei Swordmaster ein Kontakt der Schwerter beziehungsweise des Schwertes mit dem Körper registriert und in eine Reaktion umgewandelt. Zusätzlich unterscheidet das System beim Schwert zwischen Spitze und Klinge und beim Körper zwischen Kopf, Torso, Armen und Beinen. Trifft ein Hieb oder Stich eine der genannten Zonen, wird der getroffene Spieler mit einem Malus belegt. Diese Einschränkung spiegelt sich nicht wie gewöhnlich in einem Lebensbalken wieder, sondern wird über ein facettenreiches Feedback-System (Sound, Animation und Shader) kommuniziert.

Projektbeginn

Anfang Februar 2008 erstellten Markus Ambrus und Rainer Lonau das Konzept zum Bachelor-Projekt Swordmaster – das Spiel, das Ende September den Abschluss des Studiums an der Mediadesign-Hochschule in München darstellte. Nach drei Jahren und mehreren kleinen Projekten, die immer auf die unterschiedlichen Unterrichtsblöcke folgten, hatte sich ein gut eingespieltes Team gebildet: Markus Ambrus war für Charakter-Modellierung, Animationen und das Projektmanagement zuständig. Victor Linke entwarf das Interface, die Texturen und das Intro. Bettina Urban lieferte als 2D/3D-Artist Normalmaps und 3D-Objekte. Michaela Reichenberger gestaltete die Welt mit Geometrie und Texturen aus. Des Weiteren stießen im Projektverlauf externe Helfer zum Projekt dazu und steuerten unter anderem die Musik und die Soundeffekte bei.

Fokus Gameplay

Im Februar fand der einführende Unterricht zur 3D-Engine Shark 3D statt. Erste Beispiele mit Animation, Kollision und Eingabe zeigten die Funktionsweise der Entwicklungsumgebung. An einem Konzept-Wochenende entstand die Idee zum Multiplayer-Schwertkampf-Spiel. Vor allem der Hollywood-Streifen »Königreich der

Swordmaster ist ein Multiplayer-Spiel für zwei Schwertkämpfer, weitere Teilnehmer können sich als Zuschauer einklinken. Der Fokus liegt auf realistisch choreografierten Duellen mit einer genauen Kollisionserkennung.

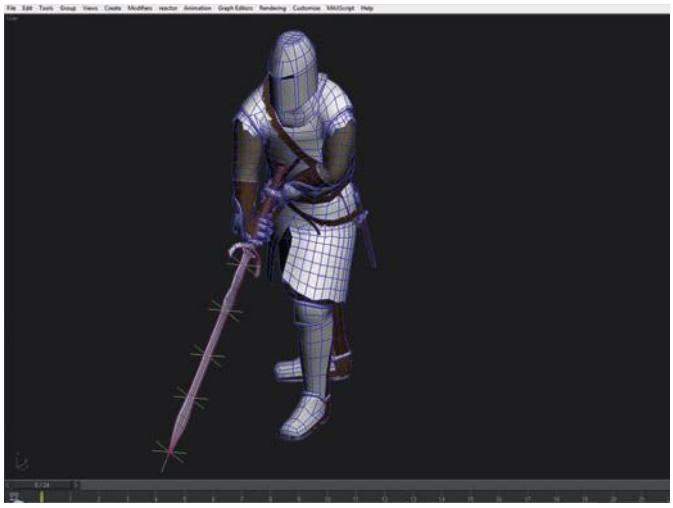


Abbildung 2: Das Schwert ist mit zahlreichen Helper-Objekten verknüpft, die im Kampf eine genaue Kollisionsabfrage ermöglichen.

Abbildung 3: Ein Screenshot aus der Shark 3D-Engine. Die eingeblendeten Quader zeigen die vielen unterschiedlichen Hitboxen an der Kämpferfigur.

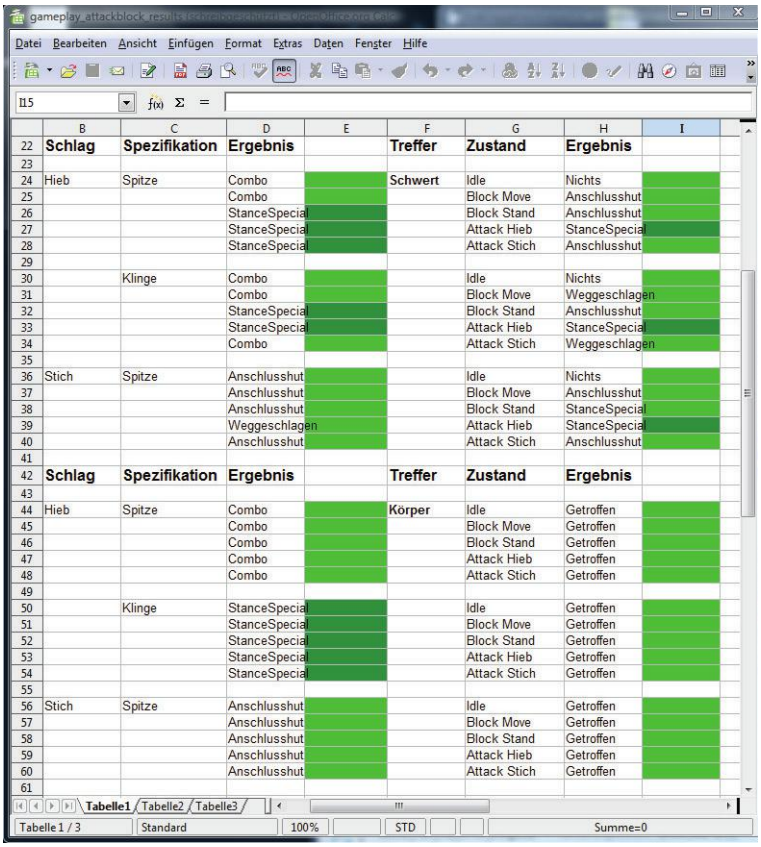
Himmel« und der darin praktizierte Kampfstil waren inspirierende Quellen. Schnell war klar, dass die Kollisionsabfrage der Schwerter und realistische Auswirkungen das Besondere am Spiel ausmachen sollten. Für den Rest der Entwicklung stand die Kollisionsabfrage im Mittelpunkt.

Pre-Production

In den Semesterferien und in den letzten Unterrichtswochen im sechsten Semester investierte das Team viel Zeit, um tiefer in die Engine einzusteigen und Tests durchzuführen. Bevor an eine Umsetzung der Spielidee gedacht werden konnte, waren diverse Fragen zu beantworten. Die größte Herausforderung stellte dabei die Kollisionserkennung dar. Deshalb stand neben der Multiplayer-Funktion und dem Animationsmanagement vor allem sie im Vordergrund. Nach einigen Tagen entstanden in den einzelnen Teilbereichen aussagekräftige Testszenarios. Im Fall der Kollisionserkennung mussten die Größe der Hitboxen getestet und mögliche Verbindungsschwankungen im Multiplayer-Modus beachtet werden. Die erfolgversprechenden Ergebnisse bildeten die Grundlage zur Realisierung des Konzepts. Im weiteren Verlauf der Pre-Production wurden Arbeitsabläufe entwickelt und unter den Teammitgliedern abgesprochen. Parallel zu den Tests analysierte das Team Haltungen und Schwerthiebe des mittelalterlichen Schwertkampfs anhand von Lehr-DVDs. Auf Grund dieser Erkenntnisse wurde das Gameplay weiter verfeinert und in Dokumenten wie »Logik der Trefferauswertung« (Abbildung 1) festgehalten. Im Spiel kann der Held eine von vier Grundhaltungen einnehmen. Dabei hält die Spielfigur das Zweihandschwert in einer typischen Schwertkampfposition. Zwei bis drei Angriffsmöglichkeiten leiten sich aus der gewählten Grundhaltung ab und stehen zur Auswahl.

Das Kollisionssystem

Das Schwert des Angreifers schwingt bei einem Treffer nicht durch den Gegner, sondern reagiert realistisch auf den Treffer. Ein Spieler kann durch die korrekte Haltung seines Schwertes oder eine gut platzierte Attacke einen gegnerischen Angriff abwehren. Ein Treffer wird registriert, wenn ein Schwert auf das andere prallt oder einen Körperteil trifft. Am Schwert der Spielfigur sind mehrere so genannter Helper-Objekte hierarchisch verlinkt (Abbildung 2). Diese bewegen sich bei Schwert-Animationen mit und ermöglichen so eine exakte Trefferabfrage. Damit Kollisionen stattfinden können, sind an der Spielfigur sowie am Schwert Hit-



	B	C	D	E	F	G	H	I
22	**Schlag**	**Spezifikation**	**Ergebnis**		**Treffer**	**Zustand**	**Ergebnis**	
23								
24	Hieb	Spitze	Combo		Schwert	Idle	Nichts	
25			Combo			Block Move	Anschlusshut	
26			StanceSpecial			Block Stand	Anschlusshut	
27			StanceSpecial			Attack Hieb	StanceSpecial	
28			StanceSpecial			Attack Stich	Anschlusshut	
29								
30		Klinge	Combo			Idle	Nichts	
31			Combo			Block Move	Weggeschlagen	
32			StanceSpecial			Block Stand	Anschlusshut	
33			StanceSpecial			Attack Hieb	StanceSpecial	
34			Combo			Attack Stich	Weggeschlagen	
35								
36	Stich	Spitze	Anschlusshut			Idle	Nichts	
37			Anschlusshut			Block Move	Anschlusshut	
38			Anschlusshut			Block Stand	StanceSpecial	
39			Weggeschlagen			Attack Hieb	StanceSpecial	
40			Anschlusshut			Attack Stich	Anschlusshut	
41								
42	**Schlag**	**Spezifikation**	**Ergebnis**		**Treffer**	**Zustand**	**Ergebnis**	
43								
44	Hieb	Spitze	Combo		Körper	Idle	Getroffen	
45			Combo			Block Move	Getroffen	
46			Combo			Block Stand	Getroffen	
47			Combo			Attack Hieb	Getroffen	
48			Combo			Attack Stich	Getroffen	
49								
50		Klinge	StanceSpecial			Idle	Getroffen	
51			StanceSpecial			Block Move	Getroffen	
52			StanceSpecial			Block Stand	Getroffen	
53			StanceSpecial			Attack Hieb	Getroffen	
54			StanceSpecial			Attack Stich	Getroffen	
55								
56	Stich	Spitze	Anschlusshut			Idle	Getroffen	
57			Anschlusshut			Block Move	Getroffen	
58			Anschlusshut			Block Stand	Getroffen	
59			Anschlusshut			Attack Hieb	Getroffen	
60			Anschlusshut			Attack Stich	Getroffen	
61								

Abbildung 1: In dieser Excel-Tabelle sind die Trefferlogik-Ausgänge verzeichnet.

Projektdaten

- fünfköpfiges Kernteam, externe Unterstützung bei Sound, Sprecher und Support
- Dauer: sechs Wochen Pre-Production, zwölf Wochen Produktionszeit
- Tools: Shark 3D, 3ds max, Photoshop, Z-Brush, Puppetshop, SVN

Projekt-Features

- vier Grundhaltungen mit acht »Hieb«-Angriffsrichtungen
- zwei Spezialhaltungen mit »Stich«-Angriffen
- vier Trefferzonen mit negativen Effekten
- drei Siegerposen mit positiven Effekten
- »Cheer«-System im Zuschauermodus, Einflussnahme auf das Kampfgeschehen
- LAN & Internet

Weiterführende Links

- Weitere Informationen zum Projekt und den Teammitgliedern: www.swordmaster-game.de
- Alles rund um die Mediadesign Hochschule und die möglichen Studiengänge: www.mediadesign.de
- Homepage der 3D-Engine von Spinor: www.shark3d.com
- Das Portal für Europäische Kampf- und Bewegungskünste. Die ausführlichen Lehr-DVDs dienten der Recherche: www.agilitas.tv

```
func void myHitHandling( int intparam )

    if intparam == 0 then
    // enemy sword_plane hit my body!

        // defender
        myHitHandlingDefender( "hit", "got_hit" ); //cmd, sound
        // attacker
        myHitHandlingAttacker( "stancespecial_playback", "hit" ); //cmd, sound

    elseif intparam == 1 then
    // enemy sword_front hit my body!

        // defender + attacker
        if m_attackerCombatInfo[2].getSub( 0, 7 ) == "special" then
            myHitHandlingDefender( "hit", "got_hit" ); //cmd, sound
            myHitHandlingAttacker( "anschlusshut", "hit" ); //cmd, sound
        else
            myHitHandlingDefender( "hit", "got_hit" ); //cmd, sound
            myHitHandlingAttacker( "combo", "hit" ); //cmd, sound
        end

    elseif intparam == 2 then
    // enemy sword_plane hit my sword!

        // defender + attacker
        if defenderDamage == "true" then
            if defenderAttackDirection.getSub( 0, 7 ) == "special" then
                myHitHandlingDefender( "anschlusshut", "hit_sword" ); //cmd, sound
                myHitHandlingAttacker( "combo", "hit_sword" ); //cmd, sound
            else
                myHitHandlingDefender( "anschlusshut", "hit_sword" ); //cmd, sound
                myHitHandlingAttacker( "stancespecial_playback", "hit_sword" ); //cmd, sou
            end
        elseif defenderBlocking == "true" then
            if defenderMoving == "false" then
                myHitHandlingDefender( "anschlusshut", "hit_sword" ); //cmd, sound
                myHitHandlingAttacker( "stancespecial_playback", "hit_sword" ); //cmd, sou
            else
                myHitHandlingDefender( "anschlusshut", "hit_sword" ); //cmd, sound
                myHitHandlingAttacker( "combo", "hit_sword" ); //cmd, sound
            end
        else
            if m_attackerCombatInfo[2].getSub( 0, 7 ) != "special" then
                myHitHandlingDefender( "idle", "hit_sword" ); //cmd, sound
```

Abbildung 4: Ein Teil des Programmcodes, der für die Trefferabfrage zuständig ist.

boxen angebracht (Abbildung 3). Die an die Proportionen der Spielfigur angepassten Hitboxen dienen nicht nur der Trefferabfrage, sondern bilden auch einen weiteren Gameplay-Layer. Der getroffene Spieler erhält nämlich einen Malus auf die verwundete Körperregion. So bedeuten etwa zwei Treffer am Kopf den Tod und somit das Ende des Kampfs. An den Armen, dem Torso und den Beinen beeinträchtigen Wunden bis zu drei Mal die Angriffsgeschwindigkeit, die Ausdauerregeneration oder die Bewegungsgeschwindigkeit des Ritters. Ein geübter Spieler kann den Gegner gezielt schwächen, dadurch Vorteile im Kampf erringen und schließlich als Sieger in die nächste Runde einziehen.

Ein Treffer – die Auswirkungen

Bei einem Treffer – also dem Kontakt eines Helper-Objekts mit einer Hitbox – wird von der 3D-Engine ein Event ausgelöst. Dieser Event wird im eigens geschriebenen Code abgefangen und verarbeitet (Abbildung 4). Um das Problem der Asynchronität bei Treffern gar nicht aufkommen zu lassen, führt Swordmaster alle Trefferabfragen nur auf dem Server durch. Bei einem Treffer müssen zuerst die beiden Spielfiguren ermittelt werden. Der getroffene Spieler erhält je nach Kampfzustand einen Befehl. Bei einem Block, beziehungsweise einem Schwert-Schwert-Kontakt gelangt der Spieler in die nächste Grundhaltung. Der Spieler wird also nicht beeinträchtigt und kann selbst die Initiative ergreifen. Wird allerdings eine Hitbox am Körper getroffen, stoppen aktuelle Animationen, eine »Getroffen«-Animation läuft ab, und der Spieler erhält den entsprechenden Malus. Der angreifende Spieler bekommt dagegen andere Reaktionen bei einem Treffer. So wird ermittelt, mit welchem Teil des Schwertes der Gegner getroffen wurde. War es die Schwertspitze, laufen die Animationen weiter. Ein Blutpartikeleffekt und Sounds geben jedoch Feedback auf einen erfolgreichen Angriff. Das Kampfsystem in Swordmaster ermöglicht dem Angreifer bei weiterführenden Schlägen die Möglichkeit schneller Anschlussangriffe. Nach dem Angriff mit der Schwertspitze ist also direkt ein weiterer Hieb möglich. Neben der Schwertspitze kann aber auch die Klinge des Schwertes einen Treffer auslösen. Dies geschieht meist im direkten Nahkampf. Beim getroffenen Spieler findet der gleiche Prozess wie oben beschrieben statt. Der Angreifer gelangt hingegen in eine spezielle Haltung. In diesen »Ochs« oder »Pflug« genannten Haltungen wird der Gegner direkt bedroht und gefährliche Stichangriffe sind möglich. Das sind Momente, die ein aggressiver, treffsicherer Spieler erzwingen und somit das Duell für sich entscheiden kann.

Markus Ambrus, Rainer Lonau

Der Backstage-Pass der Spielebranche!

JEDE AUSGABE MIT FIRMENREGISTER UND PRAKTIKUMSBÖRSE
making games MAGAZIN
04/2008 € 6,90
MAGAZIN FÜR SPIELE-ENTWICKLUNG UND BUSINESS-DEVELOPMENT
COMMUNITY MANAGEMENT
WARUM ES JEDEN BETRIFFT, WOMIT SIE EFFIZIENT STEUERN UND WIE SIE AUS LAUFKUNDEN TREUE MITGESTALTER MACHEN
DAS ENDE DER KONSOLEN
EINE PROVOKANTE ANALYSE DES DIRECTX-ERFINDERS
EXKLUSIV-INTERVIEW
IM GESPRÄCH MIT NINTENDO-LEGENDE SHIGERU MIYAMOTO
PROGRAMMIERUNG
WIE SIE IHR SPIEL PERFORMANT AUF DIRECT3D 10 PORTIEREN

Im Abo **30%** sparen!

Making Games Magazin
Das Magazin für Spieleentwicklung und BusinessDevelopment

Profi-Wissen
Von Insidern für Insider

Profi-Chancen
Immer mit Firmenregister und Praktikumsbörse

Making Games Magazin erscheint im Verlag IDG Entertainment Media GmbH, Lyonel-Feininger-Str. 26, 80807 München, Handelsregister München, HRB 116413, Geschäftsführer: York von Heimburg. Die Kundenbetreuung erfolgt durch den Making Games Magazin Kundenservice, ZENIT Pressevertrieb GmbH, Postfach 810580, 70522 Stuttgart, Geschäftsführer: Peter Staudenmaier, Tel.: 0711/7252278, eMail: shop@makinggames.de

Bestellen Sie jetzt Ihren Backstage-Pass! Ihre Vorteile: 1) 30% gespart! 2) Bequeme Zustellung, pünktlich, portofrei!

Infos & Bestellung:
www.gamestar.de/shopcode

Shop-Code 292
Shopcode merken für die Internet-Bestellung

oder
Tel.: 0711-7252 278
Mo-Fr 9-17 Uhr

4 HEFTE ☐ Ja, bitte schicken Sie mir das Making Games Magazin zum Jahresvorzugspreis von nur 19,30 €. Ich spare damit 30%. Die Lieferung erfolgt 4 x im Jahr und ist für mich vesandkostenfrei. Wenn ich 6 Wochen vor Ablauf des ersten Bezugsjahres nicht kündige, beziehe ich das Magazin ein weiteres Jahr.

1 HEFT ☐ Ja, bitte schicken Sie mir das nächste Making Games Magazin zum Einzelpreis von nur 6,90 € versandkostenfrei zu.

Firma ..... Name, Vorname .....

Straße/Postfach ..... PLZ ..... Ort .....

Telefonnummer ..... E-Mail ..... Fax .....

☐ Ich habe beruflich mit Spielen zu tun (Firma/Position) .....
☐ Ich bin interessierter Spieler
☐ Ich wünsche folgende Zahlungsweise ☐ bargeldlos und bequem per Bankeinzug ☐ per Rechnung

Bank ..... BLZ ..... Kto.-Nr. .....

8GM292A11J

Datum/Unterschrift .....

Berufschance Spiele-Industrie
Praktikumsbörse

Sie studieren Informatik, möchten aber lieber Spiele statt Büro-Software programmieren? Sie kennen sich mit Maya aus und wollen Ihr künstlerisches Talent zum Beruf machen? Hier ist Ihre Chance: Spiele-Entwickler und Publisher aus ganz Deutschland suchen Praktikanten.

Praktikanten gesucht?

Sie suchen motivierte und talentierte Praktikanten? Mit Making Games, dem Fachmagazin für Spiele-Entwicklung und Business-Development, haben Sie eine ideale Plattform für Ihre Anzeige. Schicken Sie uns Ihr Angebot an: **info@makinggames.de, Betreff: Praktikum** Der Abdruck ist kostenfrei.

PLZ-BEREICH 10000

Exozet Games
Art des Praktikums: Nach Absprache
Bereich: Maya 3D Max Modelling, Animation
Mindestdauer: 6 Monate
Ansprechpartner: Bartol Ruzic, bewerbung@exozet.com
Adresse: Rotherstraße 20, 10245 Berlin

Fraunhofer IGD Rostock
Art des Praktikums: Nach Absprache
Bereich: Prototyping, Modellierung, Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Holger Diener, holger.diener@igd-rfraunhofer.de
Adresse: J.-Jungius-Str. 11, 18059 Rostock

Fraunhofer IGD Rostock
Art des Praktikums: Nach Absprache
Bereich: User Interface Design, Charakterdesign/Illustration, Sound- Akustikdesign
Mindestdauer: 3 Monate
Ansprechpartner: Katharina Schmidt, katharina.schmidt@igd-rfraunhofer.de
Adresse: J.-Jungius-Str. 11, 18059 Rostock

Future Interactive
Art des Praktikums: Studienbegleitend
Bereich: Java-Programmierung, C++-Programmierung
Mindestdauer: Nach Absprache
Ansprechpartner: Felix Kollmann, fkollmann@futureint.de, 030 / 810 593 36
Adresse: Gartenstr. 1, 14169 Berlin

GameDuell
Art des Praktikums: Studienbegleitend, nach Absprache
Bereich: Programmierung, Marketing, Controlling/Finanzen
Mindestdauer: 6 Monate
Ansprechpartner: Tina Haile, jobs@gameduell.de, 030 / 288 768 211
Adresse: Ackerstr. 14-15, 10115 Berlin

Radon Labs
Art des Praktikums: Studienvorbereitend, studienbegleitend, Praxis-Semester, Diplom-Semester
Bereich: Programmierung
Mindestdauer: 3 Monate, besser 5-6 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstrasse 12, 12435 Berlin

Radon Labs
Art des Praktikums: Studienvorbereitend, studienbegleitend, Praxis-Semester, Diplom-Semester
Bereich: 3D-Grafik
Mindestdauer: 3 Monate, besser 5-6 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstrasse 12, 12435 Berlin

Radon Labs
Art des Praktikums: Studienvorbereitend, studienbegleitend, Praxis-Semester, Diplom-Semester
Bereich: Animation
Mindestdauer: 3 Monate, besser 5-6 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstrasse 12, 12435 Berlin

Radon Labs
Art des Praktikums: Nach Absprache
Bereich: Maya Modelling, Animation
Mindestdauer: 3 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstrasse 12, 12435 Berlin

Radon Labs
Art des Praktikums: Nach Absprache
Bereich: Programmierung (3D, AI, Physik)
Mindestdauer: 3 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstrasse 12, 12435 Berlin

Radon Labs
Art des Praktikums: Nach Absprache
Bereich: Leveldesign
Mindestdauer: 3 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstrasse 12, 12435 Berlin

Silver Style Entertainment
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Scripting (Python)
Mindestdauer: 6 Monate
Ansprechpartner: Carsten Strehse, carsten.strehse@silver-style.com
Adresse: Am Borsigturm 12, 13507 Berlin

Yager Development
Art des Praktikums: Vollzeit, auch Praktikumssemester
Bereich: Programmierung, Scripting, Grafik, Leveldesign, Produktionsassistenz
Mindestdauer: 3 Monate
Ansprechpartner: Uwe Beneke, beneke@yager.de
Adresse: Schlesische Str. 26, 10997 Berlin

Yager Development
Art des Praktikums: Vollzeit, auch Praktikumssemester
Bereich: Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Philipp Schellbach, schellback@yager.de
Adresse: Schlesische Str. 26, 10997 Berlin

PLZ-BEREICH 20000

Borgmeier Media Communication
Art des Praktikums: Praktikum für Studenten mit Schwerpunkt Marketing
Bereich: Games-PR, Corporate-PR, Marketing
Mindestdauer: 3 Monate
Ansprechpartner: Tanja Deilecke, deilecke@agentur-borgmeier.de, 040 / 413 09 60
Adresse: Rothenbaum chaussee 5, 20148 Hamburg

CNT Media
Art des Praktikums: Nach Absprache (Übernahme möglich)
Bereich: Kommunikation, Level-Design, Projekt-Planung
Mindestdauer: 3 Monate
Ansprechpartner: Cornel Hillmann, ch@cnt-media.net
Adresse: Süderstr. 159a, 20537 Hamburg

Daedalic Entertainment
Art des Praktikums: Vollzeit
Bereich: Programmierung, Grafik, Marketing
Mindestdauer: 3 Monate, besser 6 Monate
Ansprechpartner: Dominik Meyer, dm@daedalic.de, 040/ 432 612 72
Adresse: Papenreye 53, Workport Unit 2, 22453 Hamburg

dtp entertainment
Art des Praktikums: Nach Absprache
Bereich: PR, Marketing, Grafik, Test, Produktmanagement
Mindestdauer: 6 Monate
Ansprechpartner: Claas Wolter, c.wolter@dtp-ag.com, 040 / 669 910 43
Adresse: Goldbekplatz 3-5, 22303 Hamburg

House of Tales Bremen
Art des Praktikums: Vollzeit
Bereich: C++/Delphi-Programmierung, Scripting, Implementation, Integration
Mindestdauer: 3 Monate
Ansprechpartner: Tobias Schachte, tobias@house-of-tales.com
Adresse: Dobbenweg 3, 28203 Bremen

House of Tales Bremen
Art des Praktikums: Vollzeit
Bereich: 3DSMAX, Rendering, Low-Poly-Modelling
Mindestdauer: 3 Monate
Ansprechpartner: Tobias Schachte, tobias@house-of-tales.com
Adresse: Dobbenweg 3, 28203 Bremen

House of Tales Bremen
Art des Praktikums: Vollzeit
Bereich: In-Game-Dialog, Cinematics Scripting, Storytelling, Design-Dokumentation
Mindestdauer: 3 Monate
Ansprechpartner: Martin Ganteföhr, martin@house-of-tales.com
Adresse: Dobbenweg 3, 28203 Bremen

Indigo Pearl
Art des Praktikums: Nach Absprache
Bereich: PR-Agentur
Mindestdauer: 6 Monate
Ansprechpartner: Torsten Oppermann, to@indigopearl.de, 040 / 253 302 67
Adresse: Süderstr. 159a, 20537 Hamburg

Innogames
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Software-Entwicklung (PHP), Web- & Grafik-Design, Systemadministration (Linux), 3D-Modelling
Mindestdauer: 6 Wochen
Ansprechpartner: Michael Zillmer, info@innogames.de, 04141 / 776 177
Adresse: Theodor-Haubach-Weg 2, 21684 Stade

northworks
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Grafik (2D/3D)
Mindestdauer: 3 Monate
Ansprechpartner: Sascha Kaddatz, jobs@northworks.de, 040 / 781 070 75
Adresse: Stahltwiete 23, 22761 Hamburg

northworks
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Programmierung PHP
Mindestdauer: 3 Monate
Ansprechpartner: Sascha Kaddatz, jobs@northworks.de, 040 / 781 070 75
Adresse: Stahltwiete 23, 22761 Hamburg

northworks
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Programmierung Flash ActionScript
Mindestdauer: 3 Monate
Ansprechpartner: Sascha Kaddatz, jobs@northworks.de, 040 / 781 070 75
Adresse: Stahltwiete 23, 22761 Hamburg

northworks
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Community-Management (mehrsprachig)
Mindestdauer: 3 Monate
Ansprechpartner: Sascha Kaddatz, jobs@northworks.de, 040 / 781 070 75
Adresse: Stahltwiete 23, 22761 Hamburg

SFK Spielekanal Fernsehen
Art des Praktikums: Vollzeit, nach Absprache
Bereich: u.a. Game-Entwicklung und Programmierung, Game Konzeption und Design, iTV, IPTV, DVB-H im Spiele-Kontext
Mindestdauer: 3 Monate
Ansprechpartner: Claudia Marloh, claudia.marloh@spika.tv, 040 / 348 094 974
Adresse: Alter Wandrahm 15, 20457 Hamburg

SnapDragon Games
Art des Praktikums: Nach Absprache
Bereich: Grafik, (Modelling, Leveldesign, Basis Maya)
Mindestdauer: 3 Monate
Ansprechpartner: Christian von Duisburg, hr@snapdragongames.com
Adresse: Schulterblatt 58 , 20357 Hamburg

SnapDragon Games
Art des Praktikums: Nach Absprache
Bereich: Programmierung (3D, AI, Physik, Tcl)
Mindestdauer: 3 Monate
Ansprechpartner: Christian von Duisburg, hr@snapdragongames.com
Adresse: Schulterblatt 58 , 20357 Hamburg

Spacetale
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: 2D/3D-Grafik
Mindestdauer: 3 Monate
Ansprechpartner: Michael Wolf, jobs@spacetale.de
Adresse: Boppstr. 26, 55118 Mainz

PLZ-BEREICH 30000

collision-studios
Art des Praktikums: Nach Absprache
Bereich: Leveldesign
Mindestdauer: 3 Monate
Ansprechpartner: Dennis Brünig, inquiry@collision-studios.com
Adresse: Roscherstr. 5, 30161 Hannover

collision-studios
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik (zwei Plätze)
Mindestdauer: 3 Monate
Ansprechpartner: Dennis Brünig, inquiry@collision-studios.com
Adresse: Roscherstr. 5, 30161 Hannover

collision-studios
Art des Praktikums: Nach Absprache
Bereich: Scribble
Mindestdauer: 3 Monate
Ansprechpartner: Dennis Brünig, inquiry@collision-studios.com
Adresse: Roscherstr. 5, 30161 Hannover

collision-studios
Art des Praktikums: Nach Absprache
Bereich: Programmierung (zwei Plätze)
Mindestdauer: 3 Monate
Ansprechpartner: Dennis Brünig, inquiry@collision-studios.com
Adresse: Roscherstr. 5, 30161 Hannover

Reakktor Media
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik
Mindestdauer: 6 Monate
Ansprechpartner: Kirk Linke, linke@reakktor.com, 0511 / 540 980
Adresse: Arnswaldtstrasse 1030159 Hannover

Reakktor Media
Art des Praktikums: Nach Absprache
Bereich: Web-Programmierung
Mindestdauer: 6 Monate
Ansprechpartner: Kirk Linke, linke@reakktor.com, 0511 / 540 980
Adresse: Arnswaldtstrasse 1030159 Hannover

Reakktor Media
Art des Praktikums: Nach Absprache
Bereich: Leveldesign
Mindestdauer: 6 Monate
Ansprechpartner: Kirk Linke, linke@reakktor.com, 0511 / 540 980
Adresse: Arnswaldtstrasse 1030159 Hannover

Rocketbox Studios
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik
Mindestdauer: 5 Monate
Ansprechpartner: Markus Wojcik, jobs@rocketbox.de, 0511 / 935 72 80
Adresse: Vahrenwalder Str. 7, 30165 Hannover

Xybris Interactive
Art des Praktikums: Nach Absprache
Bereich: Programmierung (Java, PHP), Grafik, Webdesign (CBS)
Mindestdauer: 3 Monate
Ansprechpartner: Christian Sauer, info@xybris.com
Adresse: Apfelstr. 12a, 33613 Bielefeld

PLZ-BEREICH 40000

bhv Software
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Produktmanagement, Games, Non-Games, Edutainment, Herstellung
Mindestdauer: 3 Monate
Ansprechpartner: Dieter Dedeke, bewerbung@bhv.de, 02131 / 765 01
Adresse: Novesiastr. 60, 41564 Kaarst

bhv Software
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: PR
Mindestdauer: 3 Monate
Ansprechpartner: Saskia Heergeist, bewerbung@bhv.de, 02131 / 765 01
Adresse: Novesiastr. 60, 41564 Kaarst

Blue Byte (Ubisoft)
Art des Praktikums: Nach Absprache
Bereich: Programmierung
Mindestdauer: 6 Monate
Ansprechpartner: Sarah Samson, ssamson@ubisoft.de, 0211 / 338 000
Adresse: Adlerstr. 74, 40211 Düsseldorf

Zone 2 media
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Programmierung, 3D-Grafik
Mindestdauer: 6 Monate, nach Absprache
Ansprechpartner: Holger Offermanns, h.offermanns@zone2.de, 02150/70566-0
Adresse: Am Wasserturm 6, 40668 Meerbusch

PLZ-BEREICH 50000

Bright Future
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: C++-Programmierung
Dauer: 4 Monate, besser 6 Monate
Ansprechpartner: Gerald Köhler, jobs@brightfuture.de, 0221 / 789 821 71
Adresse: Im Zollhafen 15-17, 50678 Köln

EA
Art des Praktikums: Vollzeit
Bereich: PR, Marketing, Sales, Verwaltung
Mindestdauer: 4 Monate
Ansprechpartner: Volker Pinsdorf, www.electronic-arts.de, 0221 / 975 820
Adresse: Im Zollhafen 15-17, 50678 Köln

EA Phenomic
Art des Praktikums: Vollzeit
Bereich: 2D/3D-Grafik, 2D-Grafik, 3D-Grafik
Mindestdauer: 4 Monate
Ansprechpartner: Volker Pinsdorf, www.phenomic.de, 06132-783 50
Adresse: Binger Straße 38, 55218 Ingelheim

EA Phenomic
Art des Praktikums: Vollzeit
Bereich: Programmierung, IT-Administration
Mindestdauer: 4 Monate
Ansprechpartner: Volker Pinsdorf, www.phenomic.de, 06132-783 50
Adresse: Binger Straße 38, 55218 Ingelheim

EA Phenomic
Art des Praktikums: Vollzeit
Bereich: Qualitätssicherung
Mindestdauer: 4 Monate
Ansprechpartner: Volker Pinsdorf, www.phenomic.de, 06132-06132-783 50
Adresse: Binger Straße 38, 55218 Ingelheim

Egosoft
Art des Praktikums: Studienbegleitend oder normal
Bereich: 3D-Grafik, Design
Dauer: Mindestens 6 Monate
Ansprechpartner: Bernd Lehahn, bernd@egosoft.com
Adresse: Heidestraße 4, 52146 Würselen

Egosoft
Art des Praktikums: Studienbegleitend oder normal
Bereich: Programmierung
Mindestdauer: 6 Monate
Ansprechpartner: Bernd Lehahn, bernd@egosoft.com
Adresse: Heidestraße 4, 52146 Würselen

Independent Arts Software
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik und Animation, vorzugsweise mit Softimage XSI
Mindestdauer: 3 Monate, besser 6 Monate
Ansprechpartner: Holger Kuchling, ceo@independent-arts-software.de, 02381 / 688 292
Adresse: Münsterstr. 5, 59065 Hamm

Independent Arts Software
Art des Praktikums: Nach Absprache
Bereich: 2D-Grafik und Animation für GBA-Projekte
Mindestdauer: 3 Monate, besser 6 Monate
Ansprechpartner: Holger Kuchling, ceo@independent-arts-software.de, 02381 / 688 292
Adresse: Münsterstr. 5, 59065 Hamm

Independent Arts Software
Art des Praktikums: Nach Absprache
Bereich: Tool, 3D-Programmierung unter C++
Mindestdauer: 3 Monate, besser 6 Monate
Ansprechpartner: Holger Kuchling, ceo@independent-arts-software.de, 02381 / 688 292
Adresse: Münsterstr. 5, 59065 Hamm

Related Designs
Art des Praktikums: Studienbegleitend, Vollzeit, Praktikumssemester
Bereich: Programmierung, Grafik, Game Design
Mindestdauer: 6 Monate
Ansprechpartner: Svenja Büll, jobs@related-designs.de
Adresse: Römerpassage 1, 55116 Mainz

RTL Games
Art des Praktikums: Vollzeit
Bereich: Akquise & Produktion
Mindestdauer: 3 Monate
Ansprechpartner: Dieter Schoeller, dieter.schoeller@rtl.de
Adresse: Am Coloneum 1, 50829 Köln

PLZ-BEREICH 60000

Cooee GmbH
Art des Praktikums: Nach Absprache
Bereich: QA, Grafik, Webprogrammierung
Mindestdauer: 3 Monate
Ansprechpartner: Boris Karklins, jobs@clubcooee.com, 06301 / 703 290
Adresse: Opelstr. 10, 67661 Kaiserslautern

Deck 13
Art des Praktikums: Nach Absprache
Bereich: Character Modeler/Animator, Grafik/3D-Design, Macromedia-Flash-Design
Mindestdauer: 6 Monate
Ansprechpartner: Sarah Wrensch, jobs@deck13.de
Adresse: Baselerstraße 46, 60329 Frankfurt

Dutyfarm GmbH
Art des Praktikums: Studienbegleitend, Praxis-Semester
Bereich: Programmierung (Flash Actionscript, PHP, JSP, Datenbankprogrammierung), Grafik & Gamedesign (2D/3D), Produktmanagement, PR, QA
Mindestdauer: 6 Monate
Ansprechpartner: Daniel Schmitt, jobs@dutyfarm.com
Adresse: Oranienstraße 6, 10997 Berlin

Extra Toxic
Art des Praktikums: Studienbegleitend, nach Absprache
Bereich: Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Ivica Aracic, jobs@extratoxic.com, 069/ 408 971 43
Adresse: Hanauer Landstr. 196a, 60314 Frankfurt

Extra Toxic Online Entertainment
Art des Praktikums: Studienbegleitend, nach Absprache
Bereich: Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Ivica Aracic, jobs@extratoxic.com, 069/ 408 971 43
Adresse: Hanauer Landstr. 196a, 60314 Frankfurt

Keyfactor
Art des Praktikums: Nach Absprache
Bereich: Produktionsassistenz, Marketing & PR, Asset & Content Management
Mindestdauer: 6 Monate
Ansprechpartner: Frank Fay, jobs@key-factor.de
Adresse: Rudolf-Diesel-Str. 20a, 65760 Eschborn/Ts.

MYTIVI
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Entwicklung und Konzeption TV-Games, IPTV
Mindestdauer: 6 Monate
Ansprechpartner: Daniel Schmitt, jobs@mytivi.com
Adresse: Oranienstraße 6, 10997 Berlin

World Cyber Games
Art des Praktikums: Nach Absprache
Bereich: PR
Mindestdauer: 3 Monate
Ansprechpartner: Thomas von Treichel, tvt@ac-entertainment.com, 06152 / 985 482
Adresse: Im Neugrund 6-13, 64521 Groß Gerau

PLZ-BEREICH 70000

Feenix
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: Programmierung (Java, PHP)
Mindestdauer: 6 Monate
Ansprechpartner: Uwe Spitzmüller, uwe.spitzmueller@feenix.de
Adresse: Durmersheimerstr. 55, 76185 Karlsruhe

Gameforge
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Grafik-Design, Web-Design, Programmierung PHP, 3D-Modelling, Game Design, Community Management
Mindestdauer: 6 Monate
Ansprechpartner: Klaas Kersting, info@gameforge.de, 0721 / 120 857
Adresse: Albert-Nestler-Straße 8, 76131 Karlsruhe

SEW-EURODRIVE
Art des Praktikums: Vollzeit
Bereich: Managed DirectX Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Theres Petrasch, theres.petrasch@sew-eurodrive.de, 07251 / 751 908
Adresse: Ernst-Blickle-Str. 42, 76646 Bruchsal

Trinigy
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: Softwareentwicklung im 3D-Game-Engine-Umfeld
Mindestdauer: 3 Monate
Ansprechpartner: Felix Röken, felix.roeken@trinigy.de, 07121 / 986 993
Adresse: INKA-Businesspark, Arbachtalstr. 6, 72800 Eningen

PLZ-BEREICH 80000

Airgamer
Art des Praktikums: Vollzeit, gerne auch Praxissemester
Bereich: Web Development, Redaktion
Mindestdauer: 4 bis 6 Monate
Ansprechpartner: Katja Knecht, jobs@airgamer.de, 089 / 461 382-0
Adresse: Goethestraße 10, 80336 München

B-Alive
Art des Praktikums: Nach Absprache
Bereich: Datenmanagement, Gamedesign und assistierende Projektleitung
Mindestdauer: 6 Monate
Ansprechpartner: Arndt Schlichtig, jobs@b-alive.de
Adresse: Biberacher Straße 38, 88444 Ummendorf

B-Alive
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik, Software-Entwicklung C++
Mindestdauer: 6 Monate
Ansprechpartner: Arndt Schlichtig, jobs@b-alive.de
Adresse: Biberacher Straße 38, 88444 Ummendorf

Between Us!
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: PR und Marketing
Mindestdauer: 2 Monate
Ansprechpartner: Wolfram von Eichborn, veichborn@betweenus.de, 089 / 454 578 16
Adresse: Leonhard-Strell-Str. 11, 85540 Haar bei München

Disney Interactive
Art des Praktikums: Vollzeit
Bereich: Marketing
Mindestdauer: 6 Monate
Ansprechpartner: Christina Ettelbrück, christina.ettelbrueck@disney.com
Adresse: Kronstadter Str. 9, 81677 München

Fusionsphere Systems
Art des Praktikums: Vollzeit
Bereich: Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Martin Mayer, jobs@fusionsphere.com
Adresse: Rosenau 8, 86551 Aichach

Marchsreiter Communications
Art des Praktikums: Vollzeit
Bereich: PR
Mindestdauer: 3 Monate
Ansprechpartner: Dieter Marchsreiter, dm@marchsreiter.com
Adresse: Guldeinstr. 41a, 80339 München

ML Enterprises GmbH
Art des Praktikums: Vollzeit
Bereich: Produktmanagement, Projektmanagement, Producing
Mindestdauer: 6 Monate
Ansprechpartner: Johannes Lesser, jobs@mlenterprises.de
Adresse: Sendlinger-Tor-Platz 6, 80336 München

Take 2 Interactive
Art des Praktikums: Vollzeit
Bereich: Marketing, Produktmanagement
Mindestdauer: 6 Monate
Ansprechpartner: Jochen Till, jochen.till@take2.de, 089 / 278 221 00
Adresse: Agnesstr. 14, 80798 München

Vivendi Games Deutschland
Art des Praktikums: Vollzeit
Bereich: Marketing
Mindestdauer: 6 Monate
Ansprechpartner: Steffen Euringer, internzen@vgames.net, 089 / 999 817 0
Adresse: Fraunhoferstr. 7, 85737 Ismaning

PLZ-BEREICH 90000

Activision Deutschland
Art des Praktikums: Vollzeit, auch Praktikumssemester
Bereich: PR und Marketing
Mindestdauer: 3-6 Monate
Ansprechpartner: Bernd Reinartz, bewerbung@activision.de
Adresse: Brunnfeld 2-6, 93133 Burglengenfeld

CipSoft GmbH
Art des Praktikums: Vollzeit
Bereich: Programmierung (C++, Java, HTML, XML, Linux, ggf. Grafikprogrammierung)
Mindestdauer: 3 Monate
Ansprechpartner: Bianca Patuzzi, jobs@cipsoft.com
Adresse: Gabelsbergerstraße 11, 93047 Regensburg

Feenix
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: Programmierung (Java, PHP)
Mindestdauer: 6 Monate
Ansprechpartner: Uwe Spitzmüller, uwe.spitzmueller@feenix.de
Adresse: Durmersheimerstr. 55, 76185 Karlsruhe

HandyGames
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Grafik, Programmierung, Game- & Leveldesign
Mindestdauer: 6 Monate
Ansprechpartner: Markus Kassulke, jobs@handy-games.com, 09334 / 975 70
Adresse: i_Park Klingholz 13, 97232 Giebelstadt

qforge
Art des Praktikums: Vollzeit
Bereich: Konzeption, Programmierung (Flash/Actionscript), Systemintegration/Servermanagement (Debian Linux)
Mindestdauer: 3 Monate
Ansprechpartner: Martin Schiele, M.Schiele@qforge.de
Adresse: Südwestpark 37-41, 90449 Nürnberg

Upjers
Art des Praktikums: Vollzeit, nach Absprache
Bereich: 3D-Grafik, Programmierung, PR
Mindestdauer: 6 Wochen
Ansprechpartner: Marika Schmitt, m.schmitt@upjers.com
Adresse: Seehofstraße 46, 96117 Memmelsdorf

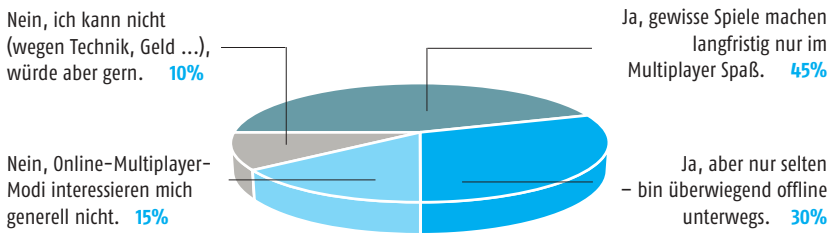
A-1000

Sproing
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Quality Assurance, Production
Mindestdauer: 5 Monate
Ansprechpartner: Johanna Schober, jobs@sproing.com, +43 / (0)1 / 604 302 838
Adresse: Fernkorngasse 10, 1100 Wien, Österreich

Ausgewählte Umfragen
Dataflow

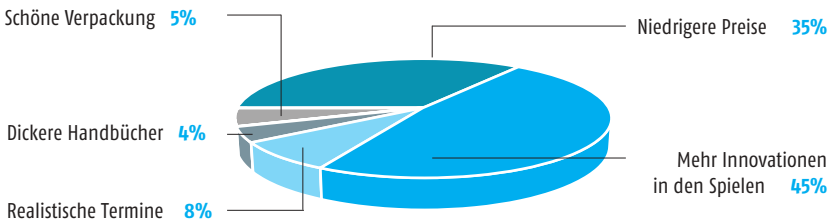
Die Redaktionen von GamePro und GameStar machen monatlich zahlreiche Umfragen unter Heftkäufern und Homepage-Usern. Eine kleine Auswahl finden Sie auf dieser Seite.

Spielen Sie Konsolentitel auch im Online-Modus?


Nein, ich kann nicht (wegen Technik, Geld ...), würde aber gern. 10%
Ja, gewisse Spiele machen langfristig nur im Multiplayer Spaß. 45%
Nein, Online-Multiplayer-Modi interessieren mich generell nicht. 15%
Ja, aber nur selten – bin überwiegend offline unterwegs. 30%

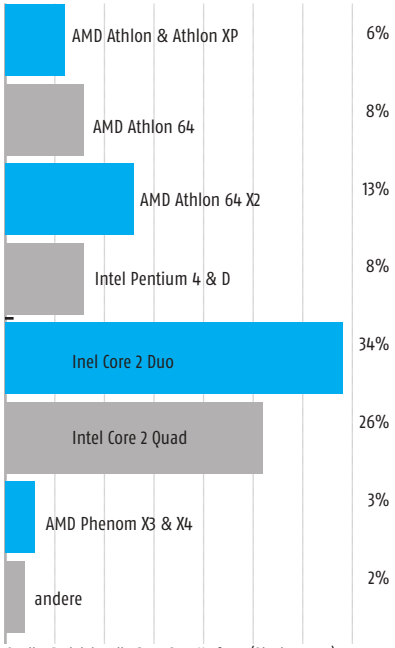
Quelle: Umfrage auf gamepro.de, 1.116 Teilnehmer.

Was wünschen Sie sich von den Publishern?



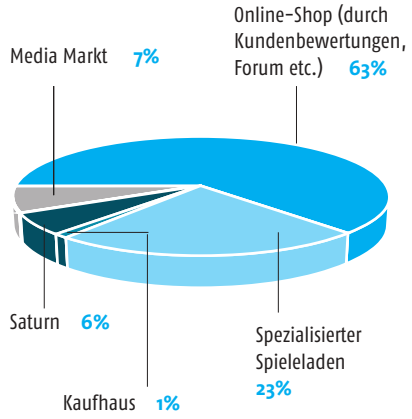
Quelle: Umfrage auf gamestar.de, 11.855 Teilnehmer.

Welchen Prozessor besitzen Sie?


AMD Athlon & Athlon XP 6%
AMD Athlon 64 8%
AMD Athlon 64 X2 13%
Intel Pentium 4 & D 8%
Inel Core 2 Duo 34%
Intel Core 2 Quad 26%
AMD Phenom X3 & X4 3%
andere 2%

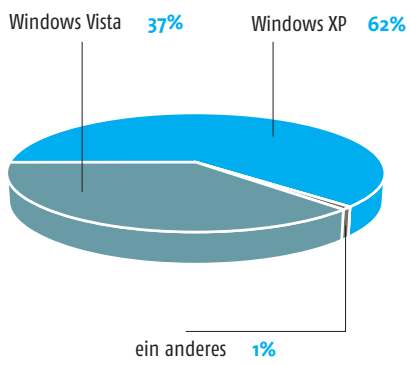
Quelle: Redaktionelle GameStar-Umfrage (Oktober 2008).

Wo fühlen Sie sich beim Spielekauf am ehesten gut beraten?



Quelle: Umfrage auf gamestar.de, 9.603 Teilnehmer.

Welches Betriebssystem nutzen Sie?



Quelle: Umfrage auf gamestar.de (23.6.2008), 4.968 Teilnehmer.

Helfen Sie uns!

Wir haben uns bemüht, ein ebenso vollständiges wie aktuelles Firmenregister zu erstellen. Damit wir uns weiter verbessern, benötigen wir Ihre Hilfe. Sie haben einen Fehler entdeckt oder vermissen Ihre Firma? Dann schreiben Sie uns: **info@makinggames.de, Betreff: Firmenregister** Bereits in der nächsten Ausgabe werden wir Ihre Änderungswünsche berücksichtigen.

Die Branche im Überblick
Firmenregister

Vergessen Sie die Auskunft! Hier finden Sie sämtliche uns bekannten Publisher, Entwickler, Dienstleister, Ausbilder und Verbände der deutschsprachigen Spiele-Industrie – von Ascaron bis Zuxxez.

PUBLISHER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
10tacle Studios	Goebelstraße 21	64293 Darmstadt	06151 / 397380	info@10tacle.com
Activision	Brunnfeld 2-6	93133 Burglengenfeld	09471 / 30690	info@activision.de
Aerosoft	Lindberghring 12	33142 Büren	02955 / 760310	info@aerosoft.de
Application Systems Heidelberg	Pleikartsförsterhof 4/1	69124 Heidelberg	06221 / 300002	kontakt@application-systems.de
Atari	Hanauer Landstraße 184	60314 Frankfurt am Main	069 / 24449100	
bad brain entertainment	Benzstraße 1b	51381 Leverkusen	02171 / 582667	team@bad-brain.com
BHV	Novesiastraße 60	41564 Kaarst-Büttgen	02131 / 76501	info@bhv.de
Capcom – CEG Interactive Entertainment	Barmbeker Straße 4b	22303 Hamburg	040 / 6965620	info@capcomeuro.de
CDV	Werner-von-Siemens-Str. 9	76646 Bruchsal	07251 / 50505601	mail@cdv.de
Codemasters	Millerntorplatz 1	20359 Hamburg	0319 / 747202	online@codemasters.de
Comport Interact!ve	Eberleinstraße 46	65195 Wiesbaden	0611 / 53169783	info@comport-interactive.com
Dartmoorsoft	Postfach 6161	32647 Lemgo	05261 / 660731	info@dartmoorsoft.com
Disney Interactive Studios	Kronstadter Straße 9	81677 München	089 / 99340191	DIS.Germany.Marketing@disney.com
dtp / Anaconda	Goldbekplatz 3-5	22303 Hamburg	040 / 6699100	info@dtp-ag.com
Eidos / Sci	Große Elbstraße 145d	22767 Hamburg	040 / 30633400	info@eidos.de
Electronic Arts	Im Zollhafen 15-17	50678 Köln	0221 / 975820	de-login@ea.com
Empire	Emmy-Noether-Straße 9	76131 Karlsruhe	0721 / 6105300	stephanie.mueller@empire.co.uk
Flashpoint	Biedenkamp 5h	21509 Glinde/Hamburg	040 / 710060	info@flashpoint.de
Frogster Interactive Pictures	Bleibtreustraße 38	10623 Berlin	030 / 2847010	info@frogster-ip.de
Halycon Media	Alte Dorfstraße 20a	23860 Klein Wesenberg	04533 / 6109-0	info@halycon.de
Jowood	Technologiepark 4a	A-8786 Rottenmann	+43 / 3614 / 29660	office@jowood.com
Koch Media	Lochhamer Straße 9	82152 Planegg/München	089 / 24245120	office@kochmedia.com
Konami	Berner Straße 103-105	60437 Frankfurt am Main	069 / 9855730	support@konami.de
Lomax Software	Postfach 1120	48482 Neuenkirchen	02551 / 919372	lomaxsoft@aol.com
Microsoft	Konrad-Zuse-Straße 1	85716 Unterschleißheim	089 / 31760	service@service.microsoft.de
MultiMediaManufaktur	Ludwigstraße 33-37	60327 Frankfurt am Main	069 / 974600	info@multimediamanufaktur.com
NBG	Brunnfeld 2-6	93133 Burglengenfeld	09471 / 70170	info@nbg-online.de
Nintendo	Nintendo Center	63760 Grossostheim	06026 / 9500	info@nintendo.de
Nokia	Heltorfer Straße 1	40472 Düsseldorf	0190 / 772222	
Novitas	Frankfurter Straße 2	38122 Braunschweig	0531 / 215340	kontakt@peppergames.de
Phenomedia	Friedrich-Ebert-Straße	44866 Bochum	02327 / 9970	Info@phenomedia.com
Rondomedia	Limitenstraße 64-78	41236 Mönchengladbach	02166 / 618660	info@rondomedia.de
RTL Games	Am Coloneum 1	50829 Köln	0221 / 7800	team@rtl-games.de
S.A.D.	Rötelbachstraße 91	89079 Ulm	07305 / 96290	office@s-a-d.de
Sega	Rablstraße 24	81669 München	089 / 22848000	info@sega.de
Sony Computer Entertainment Deutschland	Frankfurter Straße 233	63263 Neu-Isenburg	06102 / 4330	scee_presscentre@scee.net
Take 2	Agnesstraße 14	80798 München	089 / 278220	info@take2.de
TDK / Playlogic	Halskestraße 38	40880 Ratingen	01805 / 83542637	tdkgames@maxupport.de
The Games Company	Am Borsigturm 12	13507 Berlin	030 / 526853000	info@the-games-company.com
THQ	Kimplerstraße 278	47807 Krefeld	02151 / 41890	thq@thq.de
Tivola Electronic Publishing	Münzstraße 19	10178 Berlin	030 / 5363580	mail@tivola.de
Ubisoft	Adlerstraße 74	40211 Düsseldorf	0211 / 338000	info@ubisoft.de
Vivendi Universal Games	Fraunhoferstraße 7	85737 Ismaning	089 / 9998170	
Zuxxez	Rittnertstraße 36	76227 Karlsruhe	01805 / 989939	info@zuxxez.com

ENTWICKLER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
49Games	Eimsbütteler Straße 64	22769 Hamburg		kontakt@49games.de
Acony	Mönchweilerstraße 1/8	78048 Villingen-Schwenningen	07721 / 9983829	
Ascaron	Verler Straße 6	33332 Gütersloh	05241 / 96660	info@ascaron.com
B-Alive	Biberacher Straße 38	88444 Ummendorf		contact@b-alive.de
Bitfield	Rigaer Straße 56	10247 Berlin	030 / 40045181	kontakt@bitfield.de
Blue Byte	Adlerstraße 74	40211 Düsseldorf	0211 / 33800500	info@bluebyte.de
BlurredVision	Glashütter Weg 80	22844 Norderstedt		info@blurredvision.de
Boiling Point Studios	Starkenburger Straße 5	14163 Berlin		fkollmann@boilingpoint-studios.com
Buntware Games	Colmarerstrasse 9	79106 Freiburg	0761 / 2046988	contact@buntware.com
BURNS Entertainment GmbH	Marienburger Straße 26	10405 Berlin	030 / 44050281	info@burns-games.com
Bright Future	Im Zollhafen 15-17	50678 Köln	0221 / 78982171	info@brightfuture.de
Bytedefenders	Eldenaer Straße 17	10247 Berlin	030 / 42019164	info@bytedefenders.com
ByteRockers	Schönhauser Allee 36	10435 Berlin	030 / 44318610	info@byterockers.de
caipirinha games	Kiefernweg 1a	24161 Altenholz	0431 / 324540	info@caipirinhagames.de
cerasus.media	Ostseestraße 109	10409 Berlin	030 / 54982744	info@cerasus.de
Chimera Entertainment	Augsburger Straße 8	80337 München	089 / 28979940	info@chimera-entertainment.com
CipSoft	Gabelsbergerstraße 11	93047 Regensburg	0941 / 6308280	kontakt@cipsoft.com
Claw Design	Ludwigstraße 8	72474 Winterlingen	07434 / 3931	info@clawdesign.com
Cooee GmbH	Opelstraße 10	67661 Kaiserslautern	06301 / 703290	info@clubcooee.com
Cranberry Production	Podbielskistraße 166	30177 Hannover	0551 / 76170000	mail@cranberry-production.com
CreaTeam Software	Sünderup 6	24943 Flensburg	0461 / 64808	
Crenetic GmbH	Hingbergstraße 99	45468 Mülheim an der Ruhr	0208 / 4442143	info@crenetic.de
Crimson Cow	Tarpen 40	22419 Hamburg	040 / 40132614	kontakt@crimsoncow.de
Crytek	Hanauer Landstraße 523	60386 Frankfurt am Main		business@crytek.de
Daedalic Entertainment	Workport Unit 2	22453 Hamburg	040 / 432612720	info@daedalic.de
Deck13 Interactive	Guiollettstraße 48	60325 Frankfurt am Main	069 / 71671660	info@deck13.de
Destrax Entertainment	Merkurstraße 10	33739 Bielefeld		info@destrax.de
Dion Software	Hegelgasse 6/10	A-1010 Wien	+43 / 1 / 51260350	office@dion-software.com
Dutyfarm	Oranienstraße 6	10997 Berlin	030 / 30368430	info@dutyfarm.com
EA Phenomic	Binger Straße 38	55218 Ingelheim	06132 / 78350	info@phenomic.net
Egosoft	Heidestraße 4	52146 Würselen		egoinfo@egosoft.com
Eku Interactive	Saalburgstraße 104	61350 Bad Homburg	06172 / 597727	office@eku.de
Elkware	Grenzweg 1	22880 Wedel	04103 / 902090	gameseurope@infospace.com
Elocom Mobile Entertainment	Bad Essener Straße 15d	49143 Bissendorf	05402 / 9855900	elo@elocom.de
eos interactive	Rigaer Straße 56/I	10247 Berlin	030 / 40045180	info@eosinteractive.de
eWave Interactive	Dennis-Gabor-Straße 2	14469 Potsdam	0331 / 7309730	info@ewave.de
ExDream Entertainment	Glünderstraße 7a	30167 Hannover	0511 / 5152508	info@exdream.com
Exozet Games	Rotherstraße 20	10245 Berlin	030 / 2465600	games@exozet.com
Extra Toxic	Hanauer Landstraße 196a	60314 Frankfurt	069 / 48440040	info@extratoxic.com
Fakt Software	Rosa-Luxemburg-Straße 76	08058 Zwickau	0375 / 2000557	info@fakt-software.de
Fancy Bytes	Dießemer Straße 152	47799 Krefeld	02151 / 3259912	business@fancy-bytes.de
FDG Mobile Games	Kellerstraße 32	81667 München	089 / 32212356	info@fdgsoft.de
Freudenreich & Tollknaepper	Postfach 101 469	42014 Wuppertal	0202 / 2721048	mail@german-railroads.com
Funatics	Essener Straße 3	46047 Oberhausen	0208 / 43927060	friedmann@funatics.de
Fusionsphere Systems	Rosenau 8	86551 Aichach	0700 / 3874600	info@fusionsphere.com
GameDevelopment M. Molkenthin	Lerchenstraße 98	22767 Hamburg	040 / 46773207	gamedev@michael-molkenthin.de
Gameforge	Albert-Nestler-Straße 8	76131 Karlsruhe	0721 / 1208571	info@gameforge.de
gameus	Kufsteinerstraße 40	83064 Raubling		info@gameus.de
gEasy Development	Königsbacher Straße 2	67067 Ludwigshafen	0621 / 5540660	info@geasyteam.de
GIANTS Software GmbH	Birmensdorferstrasse 197	CH-8003 Zürich		mail_us@giants-software.com
Greencode	Am Hofgarten 3	53113 Bonn		info@greencode.de
Greentube	Mariahilfer Straße 47/1/102	A-1060 Wien		office@greentube.com
HandyGames	i_Park Klingholz 13	97232 Giebelstadt	09334 / 97570	info@handy-games.com
House of Tales	Postfach 1713	57278 Neunkirchen		info@house-of-tales.com
Independent Arts Software	Münsterstraße 5/Haus 4	59065 Hamm	02381 / 688292	info@independent-arts-software.de
InnoGames	Theodor-Haubach-Weg 2	21684 Stade	04141 / 776177	info@innogames.de
Island Games	Vahrenwalder Straße 7	30165 Hannover	0511 / 35319898	kontakt@islandgames.de
Keen Games	Dreieichstraße 59	60594 Frankfurt am Main	069 / 66058838	info@keengames.com
King Art	Schlüsselkorb 17/18	28195 Bremen	0421 / 322760	contact@kingart.de
Kritzelkratz 3000	Martinstraße 21	97070 Würzburg	0931 / 3536091	info@kritzelkratz.de
La Plata Studios	Papenreye 61	22453 Hamburg	040 / 42236100	contact@laplata-studios.com
Limbic Entertainment	Heinrich-Hertz-Straße 21	63225 Langen	06103 / 960357	info@limbic-entertainment.de
Madcat Interactive Software	Prager Ring 4-12	66482 Zweibrücken	06332 / 791320	info@madcat.de
marune	Wallbergstraße 16	82008 Unterhaching	089 / 66500749	info@marune.de
Master Creating	Winterhuder Weg 82	22085 Hamburg	040 / 32901223	mail@master-creating.com
Media Seasons	Liviastraße 6	04105 Leipzig	0341 / 2156752	contact@mediaseasons.com
MEF Mobile Entertainment Factory	Albin-Köbis-Straße 8	51147 Köln	02203 / 906020	info@m-e-f.net
Mobilescope	Am Meerkamp 19	40667 Meerbusch	02132 / 1370	info@mobilescope.com

ENTWICKLER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
netmin games	Philipp-Reis-Straße 6	55129 Mainz	06131 / 507896	schreiber@netmin.de
Niels Bauer Games	Sickingenstraße 37	79117 Freiburg	0761 / 66947	contact@nielsbauergames.com
northworks	Stahltwiete 23	22761 Hamburg	040 / 42236100	info@northworks.de
Novatrix	Schlüsselwiesen 19	70186 Stuttgart	0711 / 3102770	info@novatrix.de
Nuclearvision	Körnerstraße 9	38102 Braunschweig	0531 / 70189840	contact@nuclearvision.de
Nurogames GmbH	Schaafenstraße 25	50676 Köln	0221 / 3988040	info@nurogames.com
OG-Soft Productions	Zirbelstraße 48	86154 Augsburg		webmaster@og-soft.de
Outline Development	Rappenweiher 12	57074 Siegen	0271 / 3131331	webmaster@outline-development.de
Piranha Bytes	Ruhrallee 63	45138 Essen	0201 / 806720	info.neu@piranha-bytes.com
Pixon	Jaffestraße 10	21109 Hamburg	040 / 33399430	info@pixon.de
Procontis GmbH	Am Bauhof 18	64807 Dieburg	06071 / 92860	kontakt@procontis.de
Rabenauge	Kniprodestraße 18	10407 Berlin	030 / 42809100	the_contact@rabenauge.com
Radon Labs	Bouchéstrasse 12	12435 Berlin	030 /43739455	feedback@radonlabs.de
Reakktor	Arnswaldtstrasse 10	30159 Hannover	0511 / 540980	biz@reakktor.com
Redfire Software	Haarmeyers Kamp 6	49586 Neuenkirchen	05465 / 9923	contact@redfiresoft.de
Related Designs	Römerpassage 1	55116 Mainz	06131 / 5544711	info@related-designs.de
Replay Studios	Barmbeker Straße 6a	22303 Hamburg	040 / 284107830	info@replaystudios.de
Rocketscience	Robert-Bosch-Straße 7	64293 Darmstadt	06151 / 9067368	info@rocketscience.com
Rotobee	Schwedter Straße 9A	10119 Berlin	030 / 41715525	info@rotobee.com
Silent Future	Vohwinkeler Straße 51-53	42329 Wuppertal	0202 / 5143480	info@silentfuture.de
Silver Style	Am Borsigturm 12	13507 Berlin	030 / 526853000	info@the-games-company.com
Similis	Lessingstraße 6	46149 Oberhausen	0208 / 625240	info@similis.com
Sixteen Tons Entertainment	Karlstraße 3	72072 Tübingen		info@promotion-software.de
SnapDragon Games	Schulterblatt 58	20357 Hamburg	040 / 284100630	info@snapdragongames.com
Snowstep Development	Ernst-Gnoß-Straße 22	40219 Düsseldorf	0211 / 30329965	info@snowstep.com
Solimedia productions	Schlösserstraße 28	99084 Erfurt	0361 / 658550	hatmut.niemeier@solimedia.de
Spellbound	Weststraße 15	77694 Kehl am Rhein	07851 / 991660	info@spellbound.de
SpinBottle Games GmnH	Windhukweg 3	45357 Essen	0201 / 1784817 6	info@spinbottlegames.com
Spirit	Friedrichstraße 15	70174 Stuttgart	0711 / 96685510	weyrich@go4spirit.com
Springwald Software	Kurfürstenstraße 12	44623 Herne	0700 / 777464925	info@springwald.de
Sproing	Fernkorngasse 10	A-1100 Wien	+43 / 1 / 6043028	office@sproing.com
Synetic	Auf der Haar 47	33332 Gütersloh	05241 / 4009900	info@synetic.de
Ticking Bomb Games GmbH	Lerchenstraße 28	22767 Hamburg		jobs@ticking-bomb-games.de
Upjers	Seehofstraße 46	96117 Memmelsdorf	0951 / 3015835	mail@upjers.com
Vectorcom	Baumbachstraße 1	30163 Hannover		mail@vectorcom.de
Victory Soft Entertainment	1. Viertel 9	A-8171 St.Kathrein	+43 / 03179 / 8223	victory.soft@hotmail.at
VIS Games and Entertainment	Am Bitzenkamp 21d	50226 Frechen	02234 / 22983	info@vis-games.de
Vulcando Games	Werner-von-Siemens-Str. 6	86159 Augsburg	0821 / 4550834	info@vulcando-games.de
wetain	Burgstraße 10	41540 Dormagen	02133 / 882530	info@wetain.com
Xybris Interactive	Apfelstraße 12a	33613 Bielefeld	0521 / 9873035	info@xybris.com
Yager Development	Pfuelstraße 5	10997 Berlin	030 / 69597660	contact@yager.de
Zeroscale	Schwedter Straße 9a	10119 Berlin	030 / 405057920	info@zeroscale.com
zone2.media	Am Wasserturm 6	40668 Meerbusch	02150 / 705660	info@zone2.de

DIENSTLEISTER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Akamai Technologies (Datentransfer-Technologie)	Betastraße 10B	85774 Unterföhring	089 / 94006308	sales-germany@akamai.com
Anakan (Lokalisation)	Pfuelstraße 5	10997 Berlin	030 / 531420450	ana@anakan.de
Aruba Studios (Events)	Frintroper Straße 453	45359 Essen	0201 / 6131721	infos@aruba-studios.de
Atomic-Design (Grafik)	Quedlinburger Weg 27	22455 Hamburg	040 / 41265871	info@atomic-design.de
Between Us! (PR, Marketing)	Leonhard-Strell-Straße 11	85540 Haar	089 / 45457811	info@versant.com
Centroid (Motion Capturing)	Schlesische Straße 27	10997 Berlin	030 / 69807474	sven.bergmann@centroidanimation.com
Chris Noeth Illustration (Grafik)	Coburger Straße 1	97702 Münnerstadt		contact@chrisnoeth.com
Chromatrix (Mobile Gaming)	Kalkofenstraße 2	72411 Bodelshausen	07471 / 740828	info@chromatrix.com
CNT Media (3D, Animation)	Süderstraße 159a	20537 Hamburg	040 / 35085890	info@cnt-media.net
Codecult (3D-Engine)	Kemnastraße 21a	44866 Bochum	02327 / 35474	info@codecult.com
Compad Company (Merchandising)	Habsburger Straße 85	79104 Freiburg	0761 / 5853667	mz@com-pad.com
ContenTainer (Leveldesign)	Eichelkopfstraße 17	63584 Gründau	06058 / 9178150	info@contentainer.de
definitv:ton (Audio)	Lindenthalgürtel 84	50935 Köln	0221 / 8234220	mhoechsmann@definitiv-ton.de
Die Kolonie (Art, Design)	Adelheidstraße 82	65185 Wiesbaden	0611 / 5058765	contact@diekolonie.com
Dynamedion (Sounddesign)	Barbarossaring 8	55118 Mainz	06131 / 5847895	info@dynamedion.com
Effective Media (Lokalisation)	Schwarzmannstraße 8	80798 München	089 / 38998999	info@effective-media.de
Eye Rock Media (Design, Animation)	Eichsfeldstraße 62	46147 Oberhausen	0208 / 6256800	contact@eyerock.de
fadeout (Audio Design)	Austraße 6	CH-4153 Reinach	+ 41 / 79 / 2793244	info@fadeout.ch
Falko Löffler (Autor)	Lindenstraße 8a	36355 Ilbeshausen	06643 / 918577	mail@falkoloeffler.de
Fantastic Illusions (Grafik)	Beethovenstraße 40a	48317 Drensteinfurt	02508 / 985801	info@illusions.de
GD Gameduell (Spiele-Plattform)	Ackerstraße 14-15	10115 Berlin	030 / 288768211	info@gameduell.de
Global Producer (Production)	Hauptstr. 172	51143 Köln	02203 / 97722450	getintouch@global-producer.com

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Glare Studios (Grafik)	Quedlinburger Weg 27	22455 Hamburg	040 / 41265871	contact@glare-studios.de
GreenMamba-Studios (Animation)	Aloysstraße 7	48249 Dülmen	0172 / 2648745	info@greenmamba-studios.de
IGA Worldwide Ltd. (In-Game-Advertising)	Oranienburger Straße 86a	10178 Berlin	030 / 2008900	info@ingameadvertising.com
IMC (Full-Service Dienstleister)	Hauptstraße 172	51143 Köln	02203 / 97722400	info@imc-online.net
Intulo (3D, Animation)	Am Schützenholz 25	29643 Neuenkirchen	05195 / 972690	info@intulo.de
JOGO Media (In-Game-Advertising)	Graf-Adolf-Platz 15	40213 Düsseldorf	0211 / 52099491	question@jogomedia.com
Keyfactor (Consulting)	Rudolf-Diesel-Straße 20a	65760 Eschborn	06173 / 999781	contact@key-factor.de
Maurice T. Anderson (Komposition und Sounddesign)	Obertorstraße 4	63454 Hanau	0176 / 26228136	contact@maurice-t-anderson.com
ME Enterprises (Consulting)	Herrenhäuser Straße 24	30938 Burgwedel	05139 / 9599915	mmoehring@meenterprises.de
MediaLancer (Animation)	Opelstraße 10	67661 Kaiserslautern	06301 / 703280	info@medialancer.com
metricminds (Grafik)	Rüsselsheimer Straße 22	60326 Frankfurt am Main	069 / 7593380	philip@metricminds.com
Michael Bhatty Entertainment (Entwicklung, Consulting)	Bismarckstraße 24	49076 Osnabrück	0541 / 1819660	michael@bhatty.de
ML Enterprises (Consulting)	Augsburgerstraße 8	80337 München	089 / 26949648	info@mlenterprises.de
MMO hospital (Datenbank)	Wiesenkamp 22 B	22359 Hamburg	040 / 609900	mick@MMOhospital.org
Partnertrans (Lokalisation)	Leipziger Straße 28	40822 Mettmann	02104 / 172660	info@partnertrans.com
Pixelstorm (Grafik)	Tannenweg 25	24637 Bokhorst	04394 / 991458	bleicken@pixelstorm.de
Play + Smile Marketing (PR, Marketing)	Am Wasserturm 6	40668 Meerbusch	02150 / 705660	info@play-and-smile.de
Polyce (Grafik)	Hübnerstraße 4	10247 Berlin	0178 / 3592560	info@polyce.de
pro-tone (Audio, Lokalisation)	Strahlenbergerstraße 125a	63067 Offenbach am Main	069 / 2475700	info@pro-tone.de
Quality Four (Qualitätssicherung)	Behlertstraße 3a	14467 Potsdam	0331 / 2008918	info@qualityfour.de
RABCAT entertainment (Grafik)	Fernkorngasse 10	A-1100 Wien	+43 / 1 / 5237425	mail@rabcat.com
Schanz International (Consulting)	Bgm.-Oberhettinger-Str. 2a	67146 Deidesheim	06326 / 6010	info@schanzgames.com
SFK (interaktiver SpieleTV-Sender)	Alter Wandrahm 15	20457 Hamburg	040 / 348094974	info@skfernsehen.com
Spooky Cecile Media Productions (Grafik)	Chausseestraße 92	10115 Berlin	030 / 20089210	info@spooky-cecile.com
Sunny Music Musikverlag (Musikproduktion)	Hufelandstraße 13	33014 Bad Driburg	05253 / 974281	info@sunnymusic.de
The Light Works (Grafik)	Emil-Hoffmann-Straße 27	50996 Köln	02236 / 967322	info@thelightworks.com
Tiger Team Productions (Produktion)	Bahnhofstraße 27	61184 Karben	06039 / 484525	ralf@tigerteam-productions.de
ToBringAlive (Animation, Design)	Ostmarkstraße 49	48145 Münster	0251 / 392312	angerbauer@tobringalive.com
Toygardens Media (Grafik, Video)	Hubertusstraße 2	30163 Hannover	0511 / 33659800	akunze@toygardens.com
Trinigy (3D-Engine)	Arbachtalstraße 6	72800 Eningen u.A.	07121 / 986993	info@trinigy.de
Two Pi Team (Community Management, Lokalisation)	Gerresheimer Straße 9	40721 Hilden	02103 / 32276	info@two-pi-team.de
VCC Perfect Pictures (Grafik, Video)	Graf-Adolf-Straße 22	40212 Düsseldorf	0211 / 828590	o_rudolph@vcc.de
Versant (Data Management)	Wiesenkamp 22b	22359 Hamburg	040 / 609900	info@versant.com
Virgin Lands (Cinematics)	Hauptstraße 5	97332 Volkach	09381 / 71780	info@virgin-lands.com
Walk Game Productions (Produktion)	Karl-Leopold-Straße 6	76229 Karlsruhe	0721 / 48399963	info@wolfgangwalk.de
Z-Ground Illustration (Grafik)	Himmericher Straße 17a	52525 Heinsberg	02453 / 381559	jz@z-ground.com

AUSBILDER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Akademie für Medien	Balinger Straße 39A	70567 Stuttgart	0711 / 4209444	h.ehlers@amk-net.de
BiTS	Reiterweg 26b	58636 Iserlohn	02371 / 7760	patrick.streppel@bits-iserlohn.de
FH Trier	Schneidershof	54293 Trier	0651 / 8103372	
Games Academy	Rungestraße 20	10179 Berlin	030 / 29779120	info@games-academy.de
Hochschule für Gestaltung und Kunst Zürich	Ausstellungsstraße 60	CH-8005 Zürich	+41 / 43 / 4464646	hochschulsekreteriat@hgkz.de
Macromedia GmbH	Gollierstraße 4	80339 München	089 / 5441510	info.muc@macromedia.de
Mediadesign Hochschule für Design und Informatik	Lindenstraße 20-25	10969 Berlin	030 / 3992660	info-ber@mediadesign-fh.de
Mediadesign Hochschule für Design und Informatik	Werdener Straße 4	40227 Düsseldorf	0211 / 1793930	info-dus@mediadesign-fh.de
Mediadesign Hochschule für Design und Informatik	Berg-am-Laim-Straße 47	81673 München	089 / 4506050	info-muc@mediadesign-fh.de
Qantm Institute	Hoferstraße 3	81737 München	089 / 89068771	info@qantm.de
Qantm Institute	Soltauer Straße 18-22	13509 Berlin	030 / 49 86 00 54	info@qantm.de
Qantm Institute	Linke Wienzeile 130A	A-1060 Wien	+43 / 1 / 961 03 03	andreas@sae.at
SRH Fachschulen	Bonhoefferstraße 1	69123 Heidelberg	06221 / 884488	info@fachschulen.srh.de

VERBÄNDE

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
G.A.M.E.	Frintroper Straße 453	45359 Essen	0201 / 613 17 21	reichart@game-bundesverband.de
Northstar Developers	Bödekerstraße 84	30161 Hannover	0160 / 97880272	d.bruenig@collision-studios.com
ÖVUS	Aichhorngasse 14	A-1120 Wien	+43 / 1 / 810328912	office@ovus.at

Ein Tag bei ...
EA Phenomic

Unser Studio ist international ausgerichtet, deshalb auch der Bonsai-Baum im Empfangsbereich.

Zugegebenermaßen ist es für die meisten von uns nicht so wichtig, wie viel Uhr es in Shanghai ist. Aber diese Dinger sehen einfach so schick aus.

Das ist nicht unser Geschäftsführer, keine Sorge. Jobangebote findet ihr auf phenomic.de.

In der QA-Abteilung lernt man, dass Spielen richtig Stress bedeuten kann. Den Sandsack haben wir rausretuschiert.

Frank macht bei uns gerade ein Praktikum als Ziergewächs. Später möchte er mal als Baum arbeiten.

Der Herr der Prinzenrollen – unser General Manager Dirk Ringe.

Das Vorstellungsvermögen mancher Artists kann erschreckende Formen annehmen. Im Fall von 2D-Artist Ingo Sauer ist das allerdings gut so.

Nein, wir entwickeln nicht heimlich einen Shooter. Das ist nur der kleine Dienstweg zwischen Leveldesigner Markus Lubczyk und Programmierer Tobias Tost.

Der große Prophet der Planung spricht zu uns: die allgegenwärtige Excel-Liste.

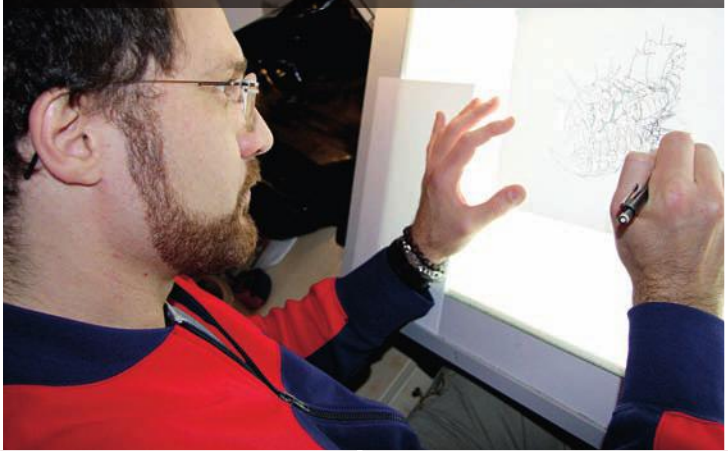
Wer von Electronic Arts gekauft wird, muss einiges auf dem Kasten haben. Die Ingelheimer Entwickler dokumentieren ihre außergewöhnlichen Fähigkeiten im Prinzenrollen-Jonglieren, Praktikanten-Züchten und Büro-Dekorieren.

Development Director Martin Löhlein und Development Manager Sebastian Cardoso jonglieren Aufgaben und Mannstunden im Planungs-Jeopardy.

FX Artist Kristoffer Lerch und Outsourcing Manager Jens Vielhaben: Wenn man wichtig genug dreinschaut, merkt keiner, dass man nur übers Zocken redet.

Unsere Artists arbeiten im so genannten »Mädchenzimmer«. Der vorne mit dem Bart ist auch ein Mädchen.

Am hellen Leuchttisch bringt Concept Artist Valerio Tosoni düstere Fantasien zu Papier.



Hier sehen wir ein Modell unseres Studiogebäudes, es fehlt nur das EA Logo an der Fassade.

Nein, keine Aktienkurse. Hier ist für jeden sichtbar, wie viele Stunden pro Sprint abgearbeitet wurden.

Für Global Community Manager Olgierd Cypra gehört es zum guten Ton, sich mit EA-Merchandise auszustatten.

Diesen Teil einer Map werden wohl nur die wenigsten sehen, aber Level Designer Michael Baur baut trotzdem einen Urlaubsort daraus.

In der nächsten Ausgabe
Making Games Vorschau

Making Games 01/2009

erscheint am 20. Februar 2009

Titelthema:

Humor in Spielen

Hand aufs Zwerchfell: Wann hatten Sie beim Spielen das letzte Mal Lachtränen in den Augen? Humorvolle Spiele werden immer seltener. Wir untersuchen, warum es so riskant ist, witzig zu sein. Und wir stellen Lösungsansätze vor, wie es trotzdem funktionieren kann. Das wird bestimmt lustig!

Weitere Themen:

- **Programmierung**
  Tipps und Tricks für eine plattformübergreifende Grafik-Engine.
- **Ein Deutscher in Japan**
  Die Geschichte einer ungewöhnlichen Entwickler-Karriere.
- **Stimmung wie im Kino**
  Welche atmosphärischen Film-Stilmittel auch in Spielen funktionieren.

Nutzen Sie uns!

Wir bieten Ihnen die Plattform für eine konstruktive Diskussion zum Standort Deutschland, nutzen Sie uns! Meinungsäußerungen und Vorschläge für Beiträge nehmen wir jederzeit entgegen. Schreiben Sie uns an **info@makinggames.de**.

IMPRESSUM

Chefredakteur
Michael Trier (verantwortlich, Anschrift d. Red.)
Chefredaktion
Markus Schwerdtel
Chefredaktion
Christian Schmidt
Redaktion
Florian Klein
Redaktion
Petra Schmitz
Layout (ltd.)
Sigrun Rüb
Layout
Jakob Scheikl
Lektorat
Anita Thiel
Praktikant
Christian Schneider

Projektleitung
Heiko Klinge
Director of Online & New Media Business
Gunnar Lott
Redaktion
Michael Graf
Redaktion (ltd.)
Daniel Visarius
Redaktion
Nino Kerl
Redaktion
Kai Schmidt
Redaktion
Fabian Siegismund
Redaktion
Hendrik Weins
Redaktion (Trainee)
Franz Philipp Dubberke

Freie Mitarbeiter dieser Ausgabe (Redaktion):
Nadja Bastawi, Gunnar Bergmann, Eddy Dew, Dr. Eyjo Gudmundsson, Steffen Kaetsch, Ralf Kornmann, Thomas Lagemann, Rainer Lonau, Sebastian Nell, Chris Noeth, Claas Oehler, Fabian Pahl, Franziska Reinhard, Dirk Riegert, Alex St. John, Teut Weidemann

Wir danken den Interview-Partnern:
David Braben, Shigeru Miyamoto

Titelgrafik: h2design.de

Assistenz:
Christine Freise, Isa Stamp, Anita Thiel, Annie Weissenberger

Korrespondenten:
Stacy Cowley (New York), Zhang Ran (Shanghai), Martyn Williams (Tokio), Peter Sayer (Paris), Sumner Lemon (Taipeh), George Dragon (Budapest), Lukas Erben (Prag)

Kundenservice

Abonnement und Einzelbestellung:
Making Games Kundenservice
ZENIT Pressevertrieb GmbH
Postfach 81 05 80, 70522 Stuttgart
Tel.: 0711 / 7252-278, Fax: 0711 / 7252-377
Österreich: Tel.: 01 / 219 55 60;
Schweiz: Tel.: 071 / 314 06-15
E-Mail: shop@makinggames.de
Web: www.makinggames.de/shop
Bezugspreise:
Einzelpreis: 6,90 Euro
Jahresabo: 19,30 Euro (beinhaltet 4 Ausgaben)
Auslandspreise u. Mengenrabatt auf Anfrage
Zahlungsmöglichkeit:
Postbank Stuttgart, BLZ 600 100 70
Konto-Nr.: 311 704
So erreichen Sie die Redaktion:
IDG Entertainment Media GmbH
Redaktion Making Games
Lyonel-Feininger-Straße 26
80807 München
Tel.: 089 / 360 86-660, Fax: 089 / 360 86-652

Anzeigenkontakt

Anzeigenleiter:
Dirk Heib (-730), dheib@gamestar.de
Sales Manager Making Games:
Nicole Müller, 089 / 120 353 23, nmueller@idg-consultant.de
Stellv. Anzeigenleitung (Print):
Klaus Maurer (-673), kmaurer@gamestar.de
Stellv. Anzeigenverkaufsleitung (Online):
Carsten Rauh (-671), crauh@gamestar.de
Junior Account Manager:
Susanne Fütterer
sfuetterer@gamestar.de, Fax: (-672)
Digitale Anzeigenannahme Print:
Andreas Frenzel (-239, ltd.)
Manfred Aumaier (-602)
Digitale Anzeigenannahme Online:
IDG Sites E-Mail Sammeladresse:
IDG_OnlineAnzeigenTechnik@idg.com
Thomas Wilms (-415)
Anzeigenpreise:
Es gilt die Preisliste gültig ab 01.01.2008

Vertrieb

Gesamtvertriebsleitung:
Josef Kreitmair (-243)
Vertriebsassistenz:
Melanie Stahl (-738)
Vertriebsmarketing:
Matthias Weber (-154, ltd.),
Stefanie Kusseler (-451), Ines Pariente (-506)

Verlag

IDG Entertainment Media GmbH
Lyonel-Feininger-Straße 26
80807 München
Tel.: 089 / 360 86-0, Fax: 089 / 360 86-501
www.idgmedia.de
Geschäftsführer: York von Heimburg
Verlagsleitung: André Horn
Veröffentlichung gemäß §8, Absatz 3 des Gesetzes über die Presse vom 08.10.1949: Alleiniger Gesellschafter der IDG Entertainment Media GmbH ist die IDG Communications Media AG, München, die 100% Tochter der International Data Group Inc., Boston, USA, ist.
Vorstand:
York von Heimburg, Keith Arnot, Pat Kenealy
Aufsichtsratsvorsitzender:
Patrick J. McGovern
Produktionsleitung: Heinz Zimmermann
Druck und Beilagen:
Mayr Miesbach GmbH, Druckerei und Verlag 83714 Miesbach

Haftung & Co.

© Copyright IDG Entertainment Media GmbH
Haftung: Für die Richtigkeit von Veröffentlichungen können Redaktion und Verlag trotz Prüfung nicht haften. Die Veröffentlichungen in Making Games erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes. Auch werden Warennamen ohne Gewährleistung einer freien Anwendung benutzt.

STOP THE ZOMBIES

GIVE EM BRAIN

We are tired of stupid zombies populating games.
Help us and give 'em brain. Sign up NOW for FREE
and download the NEW xaitment BrainPack SDKs

Set up and control complex game logic in a few steps.
Generate your perfect navigation mesh with a single click.
Create realistic behavior in no time.

Join the xaitment community now and turn your idea into a stunning prototype without paying any license cost. By signing up for free, you'll receive our complete modular AI Engine plus our world-class support. Experience the future of next gen game technology and work with the smartest AI technology available.

Contact xaitment today for more information about the BrainPack Program under brainpack@xaitment.com or visit our website www.xaitment.com

xaitment
INTELLIGENCE ENGINEERED

GIVE EM BRAIN

Autodesk Premier Solutions Provider
Media & Entertainment

weltenbauer.
software für kreative und innovatoren

IGDA Chapter Meeting Berlin - 10.12.08 - Wii Development mit Unity3D

ehrliche arbeit.

Maya 2009 - Mudbox 2009 - MotionBuilder 2009 ab Herbst verfügbar !

3D Modelling und Animation

Maya Complete 2009 - jetzt inklusive einem Jahr Pflegevertrag
3ds Max 2009 jetzt mit Entertainment Extension
Mudbox 2009

Character Animation und Motion Graphics

MotionBuilder 2009
Endorphin 2.7
CAT Toolkit 3 für 3ds Max

Middleware

Unity Game Engine für PC, Mac, Wii und iPhone
Quest3D für Serious Game Entwicklung auf PC

Rendering

V-Ray für 3ds Max und Maya

www.3dsoftware-shop.de

www.weltenbauer.com

weltenbauer. - steingasse 6 - 65183 wiesbaden - tel 0611-7342 491